

**ESTUDIO PARA LA IMPLEMENTACIÓN DE MODELOS DE MACHINE
LEARNING EN LOS MÓDULOS FISCHERTECHNIK DE LA FACULTAD DE
CIENCIAS EMPRESARIALES.**

PRESENTADO POR:

**DANIEL ANDRES CASTAÑEDA RODRIGUEZ - 1088344174
JUAN DAVID RODRIGUEZ PRIETO – 1088338393**

DIRIGIDO POR:

**JOHN ANDRES MUÑOZ GUEVARA
M.Sc. Ingeniero Industrial**

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE CIENCIAS EMPRESARIALES
PROGRAMA DE INGENIERÍA INDUSTRIAL
PEREIRA
2020**

INTRODUCCIÓN

En el presente documento se registran los hallazgos acerca de la implementación de diferentes algoritmos de machine learning como complemento a la metodología STEM (Science, Technology, Engineering , Mathematics) de FischerTechnik, mediante la cual se hace posible simular procesos industriales en un laboratorio o aula de clase a través de módulos de automatización a pequeña escala, se encuentra en esta metodología una ayuda más que suficiente, a la hora de comprobar si la implementación de dichas tecnologías emergentes en el campo de la investigación de operaciones para la optimización de procesos, son realmente aplicables, y esta aplicabilidad es realmente útil.

Para esto se realizará una simulación del modelo de Fischertechnik para el clasificador de color teniendo en cuenta valores previamente tomados de los datos y generando un nuevo conjunto de datos para mejorar la calidad del proceso.

A lo largo de la investigación se hará uso de librerías de trabajo o Frameworks para Python que permitirán tanto generar, organizar y visualizar lo datos que describen el proceso como implementar y probar distintos algoritmos de machine learning.

Tabla de Contenidos

| | | |
|--------|---|----|
| 1. | CAPITULO 1. PLAN DE INVESTIGACIÓN | 6 |
| 1.1. | TITULO | 6 |
| 1.2. | PLANTEAMIENTO DEL PROBLEMA | 6 |
| 1.2.1. | Situación problema observada | 6 |
| 1.2.2. | Síntomas y causas: situación actual | 6 |
| 1.2.3. | Formulación del problema | 7 |
| 1.2.4. | Formulación de la hipótesis | 8 |
| 1.3. | SISTEMATIZACIÓN DEL PROBLEMA | 8 |
| 1.4. | OBJETIVOS DE LA INVESTIGACIÓN..... | 9 |
| 1.4.1. | Objetivo General | 9 |
| 1.4.2. | Objetivos Específicos..... | 9 |
| 1.5. | JUSTIFICACIÓN | 10 |
| 1.6. | MARCO REFERENCIAL | 11 |
| 1.6.1. | Marco Teórico | 11 |
| 1.6.2. | Marco Conceptual | 32 |
| 1.6.3. | Marco Temporal..... | 35 |
| 1.6.4. | Marco Espacial..... | 36 |
| 1.7. | METODOLOGÍA DE LA INVESTIGACIÓN | 36 |
| 1.7.1. | Diseño Del Estudio | 36 |
| 1.7.2. | Metodología | 38 |
| 1.7.3. | Fuentes de información:..... | 39 |
| 2. | CAPITULO 2. OBTENCIÓN DE DATOS | 41 |
| 2.1. | FISCHERTECHNIK (CONTEXTUALIZACIÓN)..... | 41 |
| 2.2. | ELEMENTOS | 41 |
| 2.2.1. | Controlador | 42 |
| 2.2.2. | Actuadores..... | 43 |
| 2.2.3. | Sensores | 46 |
| 2.2.4. | Robot clasificador por colores | 49 |
| 2.2.5. | Asignación De Entradas Y Salidas | 51 |
| 2.3. | RUTINA DEL ROBOT CLASIFICADOR DE COLOR | 54 |
| 2.4. | OBTENCIÓN DE LOS DATOS | 57 |
| 2.5. | CALIBRADOR..... | 60 |
| 2.6. | GENERACIÓN DE DATOS SIMULADOS..... | 63 |
| 3. | CAPITULO 3. MODELOS DE IA Y RESULTADOS | 64 |
| 3.1. | MODELOS DE MACHINE LEARNING | 64 |
| 3.1.1. | Regresión Logística..... | 64 |
| 3.1.2. | SVM (Maquina de Vectores de Soporte) | 65 |
| 3.1.3. | Clasificador de K Vecinos Cercanos..... | 67 |
| 3.1.4. | Clasificador de Procesos Gaussianos (Gaussian Process Classifier) | 67 |

| | | |
|--------|---|----|
| 3.1.5. | Clasificador de Arboles de Decisión..... | 67 |
| 3.1.6. | Clasificador de Bosque Aleatorio | 68 |
| 3.1.7. | Clasificador de Perceptrón Multicapa..... | 68 |
| 3.1.8. | Clasificador Bayesiano Ingenuo | 68 |
| 3.2. | CRITERIOS DE EVALUACIÓN PARA LOS MODELOS..... | 69 |
| 3.2.1. | Precisión..... | 69 |
| 3.2.2. | Sensibilidad..... | 70 |
| 3.2.3. | F1 Score | 70 |
| 3.3. | APLICACIÓN DE MODELOS..... | 70 |
| 4. | CONCLUSIONES | 74 |
| 5. | RECOMENDACIONES | 75 |
| 6. | BIBLIOGRAFIA | 76 |

Tabla de Ilustraciones

| | |
|---|----|
| Ilustración 1. Componentes de un Problema..... | 14 |
| Ilustración 2. Perceptrón | 23 |
| Ilustración 3. Demostración Bidimensional del efecto de las capas ocultas | 25 |
| Ilustración 4. TXT Controller | 42 |
| Ilustración 5. Válvula Solenoide | 43 |
| Ilustración 6. Minimotor de 6-9V | 44 |
| Ilustración 7. Compresor | 45 |
| Ilustración 8. Bombilla Insertable | 45 |
| Ilustración 9. Pulsador..... | 46 |
| Ilustración 10. Conexiones del Pulsador..... | 47 |
| Ilustración 11. Foto Transistor. | 47 |
| Ilustración 12. Sensor de Color..... | 48 |
| Ilustración 13. Robot Clasificador por color..... | 50 |
| Ilustración 14. Circuito Neumático. | 51 |
| Ilustración 15. Conexión TXT Controller para robot clasificador por color | 54 |
| Ilustración 16. Programación en RoboPro. | 57 |
| Ilustración 17. Distribución por color. | 58 |
| Ilustración 18. Matriz de confusión 1. | 59 |
| Ilustración 19. Matriz de Confusión 2. | 59 |
| Ilustración 20. límites de umbral calibrados desde valores altos. | 60 |
| Ilustración 21. Límites de umbral calibrados desde valores bajos..... | 61 |
| Ilustración 22. Matriz de confusión para los valores con límites de umbral calibrados. | 62 |
| Ilustración 23. Dispersión de colores en el plano. | 63 |
| Ilustración 24. Resultados para regresión logística..... | 71 |
| Ilustración 25. Resultados para SVC. | 72 |
| Ilustración 26. Resultados de modelos complejos. | 73 |

CAPITULO 1. PLAN DE INVESTIGACIÓN

1.1.TITULO

Estudio para la implementación de modelos de machine learning en los módulos FischerTechnik de la facultad de ciencias empresariales.

1.2.PLANTEAMIENTO DEL PROBLEMA

1.2.1. Situación problema observada

La confiabilidad y el análisis de los datos es un elemento crucial a la hora de tomar decisiones significativas sobre el panorama empresarial, generalmente estos datos son y han sido tomados con una intervención humana evidente, lo que trae implícito consigo mismo un factor de duda o un alto grado de error.

1.2.2. Síntomas y causas: situación actual

Todo se remonta a la época de la industrialización, en el momento en el que la transformación de materias primas, economía, sociedad y tecnología empezó su acelerado crecimiento y marcó un paso entre una economía rural basada en el comercio y la agricultura a una economía de carácter urbano, industrializada y mecanizada, Con esto, la revolución industrial marca un punto de inflexión en la historia permitiendo así para inicios del siglo XIX una multiplicación notable en la riqueza y en la renta per cápita de los países que en ese momento se encontraban a la vanguardia de la industria como era el caso de los países pertenecientes al continente Europeo y posteriormente se ve reflejada en Norte América (Lucas, 1996).

Actualmente se habla de cuatro momentos históricos que cambiaron el rumbo de las esferas económicas y sociales de los países desarrollados, estos momentos se denominan las cuatro revoluciones industriales.

El crecimiento y el avance de las tecnologías emergentes que pueden ser aplicables a la industria, se están enfocando cada vez más en el uso de los datos como lo es el caso de la inteligencia artificial, IoT (Internet de las cosas) o el Blockchain (Ernst & Young LLP, 2019), esto genera implícitamente la necesidad de asegurar la calidad de los datos y a su vez de encontrar mecanismos dentro de la industria que fortalezcan el proceso de recolección, manejo y análisis de los datos para la toma de decisiones. Basados en lo anterior y haciendo énfasis en el latente error humano dentro de la medición, tratamiento y análisis de los datos, se hace casi que necesaria la implementación de tecnologías actuales que ayuden a disminuir al máximo dichos errores humanos además de agregar el mayor valor posible a los datos suministrados y analizados, entre dichas tecnologías y/o herramientas se cuenta con el análisis estadístico, algoritmos de aprendizaje automático para los datos tomados, investigación de operaciones, entre otros.

Debido a la inminente llegada de la industria 4.0 a diferentes áreas dentro de los procesos productivos, y como propuesta de valor en respuesta a los crecientes avances tecnológicos, es necesario vincular el uso de nuevas tecnologías en la educación superior (ADEC, 2017) como lo es el caso de la metodología STEM.

1.2.3. Formulación del problema

¿Es posible aplicar Modelos de Machine Learning a herramientas de enseñanza (STEM) como los módulos de automatización y electroneumática de Fischertechnik?

1.2.4. Formulación de la hipótesis

Es posible aplicar modelos de machine learning como complemento al funcionamiento de los módulos de FischerTechnik de la facultad de ciencias empresariales.

1.3.SISTEMATIZACIÓN DEL PROBLEMA

¿Puede la aplicación de modelos de Machine Learning complementar las herramientas de enseñanza STEM basadas en los módulos de automatización y electroneumática Fischertechnik para la simulación de procesos productivos en la ingeniería industrial?

¿Qué indicadores repercuten significativamente para la aplicación de los modelos de Machine Learning planteados para los módulos de automatización y electroneumática de simulación de procesos?

¿Cuáles son los tipos de modelos de Machine Learning que diagnostican un mejor ajuste a los resultados de los casos planteados en los módulos de automatización y electroneumática del FischerTechnik?

¿Alimentar estos modelos de Machine Learning con herramientas de simulación de procesos o análisis estadístico genera mejores resultados con respecto a los casos originalmente planteados?

1.4.OBJETIVOS DE LA INVESTIGACIÓN

1.4.1. Objetivo General

Realizar un estudio para la implementación de modelos de machine learning en los módulos FischerTechnik de la facultad de ciencias empresariales.

1.4.2. Objetivos Específicos

- Evaluar la viabilidad técnica de los módulos Fischertechnik disponibles en el laboratorio para la aplicación de herramientas de machine learning.
- Crear un módulo de prueba mediante la utilización de los componentes Fischertechnik que permita simular un ambiente industrial el cual permita recopilar información necesaria para la aplicación de herramientas de machine learning.
- Probar modelos de machine learning con los datos e indicadores obtenidos en el proceso de simulación con el módulo Fischertechnik, para medir su nivel de ajuste y su impacto en los procesos simulados.
- Comparar los resultados obtenidos por los modelos de machine learning, que permitan determinar la diferencia de predicciones entre los modelos alimentados por datos simulados y los datos tomados de los módulos Fischertechnik.

1.5. JUSTIFICACIÓN

En lo que a la Inteligencia Artificial respecta, existen demasiados modelos aplicables al estudio de procesos por lo que la presente investigación no pretende refutar o reafirmar ninguno de ellos, por el contrario, se pretende dar uso a ciertos algoritmos ya creados para la revisión y el análisis de los datos, también se hace uso de modelos matemáticos para la resolución de problemas genéricos y prototípicos, estos problemas también pueden ser solucionados utilizando métodos heurísticos, que consisten en la formalización de reglas y herramientas las cuales fundamentan las bases de la solución metódica de un problema o de un conjunto de problemas, estos constituyen los pilares de los algoritmos, que gracias al avance de la ciencia se ven afectados por una rama de la inteligencia artificial que intenta simular la forma de aprendizaje humano, El *Machine Learning*, o aprendizaje de máquina.

Ahora bien, si muchos de los algoritmos o métodos heurísticos nacidos desde la investigación de operaciones son directamente aplicables en diferentes fases de los procesos industriales, a través de esta investigación se pretende evaluar si los algoritmos de machine learning derivados de estos métodos o reglas heurísticas, pueden ser aplicados también a dichos procesos, en este caso simulados a través de distintas herramientas que se irán aplicando durante el desarrollo de la presente investigación.

Con la realización del presente trabajo, se espera generar una incursión en el campo de la inteligencia artificial y del machine learning, que permita a estudiantes y docentes de la facultad conocer, fortalecer y practicar esta temática mediante el uso de los equipos Fischertechnik dispuestos en el laboratorio.

1.6.MARCO REFERENCIAL

1.6.1. Marco Teórico

Actualmente se habla de cuatro momentos históricos que cambiaron el rumbo de las esferas económicas y sociales de los países desarrollados, estos momentos se denominan las cuatro revoluciones industriales, las cuales se definen a través de las invenciones tecnológicas, como es el caso de las primeras tres revoluciones con la máquina a vapor (Overton, 1952) para la primera revolución, la electricidad y el motor de combustión para la segunda revolución y por último la creación del internet en la tercera revolución abriendo paso a grandes avances en telecomunicaciones y robótica.

Con la integración de la tecnología digital a las empresas manufactureras o de transformación, se busca generar valor en la industria por medio de cuatro maneras: conectividad mejorada, automatización de tareas manuales, toma de decisiones mejorada e innovación de productos o servicios (Olanrewaju y Willmott, 2013). Estas variables componen los pilares fundamentales de la denominada cuarta revolución industrial o industria 4.0, dicha época parte desde los procesos de manufactura y distribución, y no solo los conecta si no que crea un ambiente de comunicación, análisis y toma de decisiones basados principalmente en los datos. (Rao & Prasad, 2018)

Conocida también como la era de la información, la **industria 4.0** busca implementar elementos y herramientas de SMART Manufacturing, IoT (Internet de las cosas), Inteligencia artificial, Automatización y Robótica, para tomar, almacenar, manejar, analizar e incluso predecir comportamientos a partir de datos tomados a lo largo de todo el proceso. (Sniderman, Mahto, & Cotteleer, 2016)

La **automatización industrial** es una de las primeras aplicaciones claras de la Industria 4.0 ya que se le está enseñando a la máquina a realizar actividades que son repetitivas para los seres humanos, cuando se habla de automatización de la industria se debe hablar de las líneas de ensamble que consiste en una secuencia ordenada de tareas que realizan una cantidad de operaciones para transformar materia prima en un producto, al conjunto de tareas se le denomina proceso, la industria automotriz fue la primera en darnos un ejemplo claro de una línea de ensamble cuando Henry Ford realizó modificaciones dentro de su fábrica para aumentar el ritmo de producción de automóviles modelo T.

Uno de los elementos tecnológicos más importante de los últimos años es la internet y es importante principalmente como potenciador de la evolución, según Dave Evans (2011) *“Si evolucionamos es porque nos comunicamos”* esto se traduce de otra forma a que el hecho de estar conectados con los demás permite que la humanidad avance mucho más rápido, como se evidenció con la llegada del iPhone presentado por el director general de Apple, Steve Jobs, el 9 de enero de 2007 en la conferencia Macworld, la conectividad de elementos aumentó alarmantemente, para el año 2010 se estimaba un total de 12,5 mil millones de dispositivos conectados comparado con los 6,8 mil millones de personas en población total para el mismo año (Evans, 2011) cuya relación es de 1,84 Dispositivos por persona, desde ese momento se empezó a hablar del concepto de **“Internet de las Cosas”**, con la conectividad de los objetos y con el creciente aumento de elementos conectados por persona, se empiezan a registrar grandes volúmenes de datos a la red.

Al aumentar la conectividad de las personas aumenta el ingreso de datos de estas a la red, con estos datos se pueden realizar pronósticos y a partir de estos conjuntos de datos. uno de los principales problemas es cuando el volumen de datos se vuelve muy grande y el manejo y análisis de estos se hace difícil, de ahí nace el concepto de **Big**

Data, el cual corresponde a una estructura virtual que permite almacenar, procesar y analizar datos de una forma alojada como lo es la computación en la nube por dar un ejemplo. Partiendo del principio de que una máquina pueda realizar labores tal y como las realiza un ser humano, nace la Inteligencia Artificial y valiéndose del uso de diferentes algoritmos o modelos matemáticos a través de los cuales se ingresan algunas reglas o restricciones que delimitan el problema a solucionar o la actividad a ejecutar.

Para una máquina ser considerada inteligente esta debe poseer algunas características particulares que le ayuden a pasar o superar la prueba de Turing (García Serrano, 2017), que consiste a grandes rasgos en poner la máquina a interactuar con un humano o usuario y si este no es capaz de reconocer si se trata de una máquina u otro humano, se puede decir que la máquina es inteligente.

El uso de herramientas de estadística e Informática para el tratamiento de grandes volúmenes de datos está teniendo muy buenas repercusiones en los últimos tiempos, dependiendo del tipo de los datos se tienen diferentes modelos algorítmicos para caracterizar y manipular dichos datos.

Modelos de optimización

Los modelos de optimización utilizados para dar con respuestas óptimas varían dependiendo del tipo de problema, en la literatura existen varios casos prototípicos como lo son el TSP (Problema del agente viajero), El problema de la mochila (carga) y VRP (Problema de Ruteo de Vehículos) los cuales no se abordarán en esta sección pero que son necesarios como punto de partida para entender el comportamiento de los diferentes algoritmos de búsqueda para los cuales es necesario realizar un estudio.

Un **Problema de optimización** requiere una distinción conceptual específica dentro de sus problemas a tratar iniciando por la creación del “**modelo matemático**” que especifica como es la relación entre sus variables, existen modelos tanto lineales como no lineales y cada uno de ellos tiene una concepción o una idea distinta acerca de la relación de sus variables, un modelo requiere unas entradas, inputs o “ X ’s” que son los parámetros que se observan del mundo real y a través de la interacción dentro del modelo se obtiene una salida, output o “ Y ”.

Así mismo es necesario tener en cuenta un **objetivo** que es quien dictamina cuando se ha solucionado o se da por concluido el problema el cual está más relacionado con el contexto real del problema y se transfiere al modelo matemático.

Por último, es necesario evaluar que tan buena es la solución con respecto al problema y sus demás soluciones para esto es necesario tener una **función de evaluación** que permita definir un criterio para saber no solamente si el problema ha sido resuelto sino también si se ha hecho de la mejor manera posible.

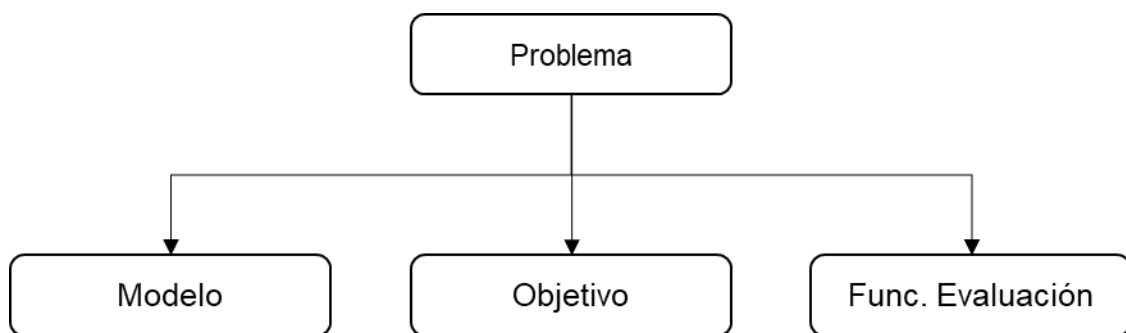


Ilustración 1. Componentes de un Problema.¹

¹ Fuente: Elaboración Propia

Existen varios tipos de modelos en lo que a inteligencia artificial respecta, partiendo desde modelos muy sencillos de **búsqueda no informada**, como lo es el caso de los **árboles de decisión**, que nacen de los grafos y sus aplicaciones para **búsqueda en amplitud** y **búsqueda en profundidad**, las cuales son formas de navegar a través de la jerarquía de los grafos.

Paralelo a esto se encuentran diferentes modelos de **búsqueda informada** la cual hace uso de la información que provee naturalmente el problema, haciendo necesario la aplicación de una función o criterio que le permita navegar de una mejor manera dentro de las posibles soluciones al problema, guiándose a partir de heurísticas como el *Backtracking* y el *A** (*A star*) así como métodos iterativos o de *búsqueda local* que buscarán llegar a una solución óptima a través de pequeñas iteraciones efectuadas a un estado **base** o estado inicial, formando así toda una **vecindad** o localidad de diferentes soluciones, esperando que cada nuevo estado resultante o **estado vecino** se acerque más a la solución.

Dentro del grupo de algoritmos de búsqueda local se encuentran los constructivos voraces o *Greedy* tales como *Dijkstra* o *Clarke & Wright*, así como también los algoritmos de *Hill climbing*, *simulated annealing*, *búsqueda tabú* y *algoritmos genéticos* por mencionar los más conocidos. (García Serrano, 2017).

Los sistemas expertos o también llamados **sistemas basados en reglas** son los precursores del razonamiento en las máquinas y son una clara muestra de lo que puede pasar cuando se almacenan en memoria anteriores resultados de la aplicación de reglas heurísticas, los **Sistemas Basados en Conocimiento**, están compuestos de dos partes principales, la primera, **la base del conocimiento** que se encarga de almacenar la información necesaria sobre el dominio de aplicación para poder ser capaz de llegar a conclusiones, y la segunda, el **motor de inferencia** que se encarga de razonar a

partir de una cuestión, buscando información en la base de conocimiento y relacionándola hasta conseguir obtener una conclusión coherente con el problema planteado.

La información dada por cierta o demostrada dentro del dominio de aplicación pueden ser hechos previamente introducido o hechos inferidos a partir de reglas las cuales permiten relacionar los hechos para inferir hechos nuevos. Dichas reglas generalmente tienen la forma de:

Antecedente \rightarrow Consecuente

La cual se asemeja a la forma IF/THEN de los bloques condicionales en una línea de código, o ilustrado de otra manera:

***SI** algo es cierto **ENTONCES** ocurre una acción.*

Existen dos estrategias de búsqueda que utiliza el motor de inferencia para navegar dentro de la base del conocimiento, la más usada es la estrategia de **encadenamiento hacia adelante** con la cual el motor identifica las reglas en las que se cumple un antecedente para obtener el consecuente, caso contrario ocurre con el **encadenamiento hace atrás**, la cual parte de un consecuente que quiere ser demostrado al cual se van encadenando los antecedentes de las reglas. Un caso sencillo para interpretar los antecedentes y los consecuentes es verlos como los síntomas y la patología respectivamente, la estrategia de encadenamiento hacia adelante buscará una posible enfermedad a través de distintos síntomas mientras que su contraria buscará los síntomas para demostrar la enfermedad.

Cuando no se pueden llegar a consecuentes de una manera binaria, se utilizan los **sistemas difusos** los cuales funcionan a través de reglas booleanas continuas, que establecen un umbral de aceptación como lo son por ejemplo “Mayor qué”, o “menor qué” Los cuales determinarán si un consecuente es válido para una regla si es mayor o menor que el umbral y no solamente si es o no igual al umbral, como lo es el caso de un termostato que a través de una configuración “reglas” busca mantener la temperatura a un nivel específico accionando la calefacción cuando la temperatura se encuentra por debajo del umbral o disminuyendo la temperatura ambiente cuando este detecta temperaturas por encima del umbral, de aquí nace la **inferencia difusa** la cual puede tomar variables de forma lingüística (tales como frío o caliente) definidas a través de los siguientes atributos:

Nombre de la variable: Descripción de la variable (temperatura).

Valores lingüísticos: Estados posibles de la variable (frío o caliente).

Universo del discurso: Valores que puede tomar la variable, relacionado con lo real, por ejemplo, la temperatura de un horno puede registrar desde 0 hasta 300 grados.

Función semántica: Asigna una definición a los valores tomados por la variable, generando una relación entre los datos tomados y su valor lingüístico.

Al tener definidas las variables se aplica el proceso de inferencia que consiste en los siguientes pasos:

Evaluación del antecedente: Se buscan las reglas dentro de la base de conocimiento. Obtención de conclusión para cada regla: al evaluar cada regla dentro de la base de conocimiento se obtiene un consecuente. Agregar conclusiones: los consecuentes deben ser agregados a la base de conocimiento para posteriormente ser nuevamente evaluados.

Nitidificación: Los consecuentes deben ser traducidos o transformados a un valor final de salida.

- **Aprendizaje:** La capacidad de una maquina inteligente no se encuentra solamente reducida al razonamiento, también requiere de un aprendizaje que le permita, a partir de información obtenida del exterior, modificar reglas o parámetros que le permitan ejecutar de una mejor manera una acción, generando abstracciones de los datos obtenidos y patrones o clasificaciones de diferentes hechos.

Existen diferentes técnicas que permiten realizar clasificaciones, entre estas se encuentra las clasificaciones basadas en modelos probabilísticos (**clasificación probabilística**) que parte de un espacio muestral de eventos posibles denotado con la letra Ω el cual posee un tamaño o cardinal y un subconjunto denominado suceso, que es a lo que se quiere hallar la tendencia de ocurrencia, a partir de dos sucesos diferentes (A y B) se pueden construir dos nuevos sucesos.

$A \cap B$: La intersección entre dos sucesos son aquellos valores favorables para ambos, pertenecen al suceso A y al suceso B

$A \cup B$: La unión entre dos sucesos son aquellos valores favorables para cualquiera de los dos, pertenecen al suceso A, al suceso B o a ambos.

Cuando se quiere evaluar la probabilidad de que un suceso ocurra se calcula con la **regla de Laplace** donde:

$$P(A) = \frac{Card(A)}{Card(\Omega)}$$

Debido a que la probabilidad siempre será un valor entre 0 y 1, existe la **probabilidad complementaria** la cual se define de la siguiente manera.

$$P(A^c) = 1 - P(A)$$

- **Probabilidad condicionada:** Se interpreta como la probabilidad del suceso A sabiendo que se da el suceso B (*A dado que B*):

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

Despejando la intercepción queda que:

$$P(A \cap B) = P(A|B) * P(B)$$

- **Independencia:** Cuando se dice que dos sucesos son independientes, la ocurrencia de uno no afecta directamente al otro, se cumple que:

$$P(A \cap B) = P(A) * P(B)$$

$$P(A|B) = P(A)$$

$$P(B|A) = P(B)$$

- **Teorema de las probabilidades totales:** Si se divide el espacio muestral en n partes tal que $\Omega = \{ A_1, A_2, A_3, \dots, A_n \}$, y siendo B cualquier suceso, se tiene que:

$$P(B) = P(B \cap A_1) + P(B \cap A_2) + \dots + P(B \cap A_n)$$

El cual se expresa de la siguiente manera:

$$P(B) = P(B|A_1) * P(A_1) + P(B|A_2) * P(A_2) + \dots + P(B|A_n) * P(A_n)$$

- **Teorema de Bayes:** Teniendo un espacio muestral dividido en n partes tal que $\Omega = \{A_1, A_2, \dots, A_i, \dots, A_n\}$, conociendo la probabilidad de cada una de las particiones A_i y la probabilidad del suceso B dado cada A_i , se puede calcular $P(A_i|B)$ de la siguiente manera:

$$P(A_i|B) = \frac{P(A_i \cap B)}{P(B)}$$

Lo cual se puede expresar de la siguiente manera:

$$P(A_i|B) = \frac{P(B|A_i)}{P(B|A_1) * P(A_1) + P(B|A_2) * P(A_2) + \dots + P(B|A_n) * P(A_n)}$$

- **Clasificador Bayesiano Ingenuo:** Es un clasificador probabilístico basado en la aplicación del teorema de Bayes. Se denomina clasificador “ingenuo” debido a que se asume que las variables a clasificar son totalmente independientes los unos de los otros. Se expresa el teorema de Bayes para el clasificador de la siguiente manera:

$$P(causa|efecto) = \frac{(P(efecto|Causa) * P(Causa))}{P(efecto)}$$

Tomando como ejemplo la causa como un “enfermedad” y el efecto como un “síntoma” para realizar la probabilidad de ser diagnosticado con dicha enfermedad dado el síntoma. Generalizando un poco la ecuación para diferentes efectos, se dispone de la siguiente manera.

$$P(causa|efecto_1, efecto_2, \dots, efecto_n) = P(causa) * \prod_i P(efecto_i|causa)$$

Con lo anterior se pretende que, dada una serie de efectos, se pueda clasificar y, por ende, encontrar la causa que los provoca. Teniendo en cuenta cómo funciona la capacidad de inferencia como generador de aprendizaje de una máquina es momento de observar las categorías y aplicaciones del aprendizaje de máquina, el **Aprendizaje Supervisado**, el **Aprendizaje no Supervisado** y el **Aprendizaje por refuerzo** los cuales se definirán a continuación:

El **aprendizaje supervisado** consiste en tener un conjunto de datos con su respectiva etiqueta o categoría y dividir el conjunto en un porcentaje de entrenamiento y en un porcentaje de prueba siendo representado generalmente en un 80/20 o en un 90/10 respectivamente. El aprendizaje supervisado busca llevar un modelo a su mejor solución partiendo de un conjunto de datos con etiquetas, aprendiendo de dichas etiquetas y extrapolando el conocimiento a nuevos casos para los cuales no se cuentan con etiquetas o categorías respectivas, algunos de los algoritmos de aprendizaje supervisado son:

- **KNN:** k-Nearest Neighbors, busca los vecinos cercanos a través de técnicas iterativas
- **Regresión Lineal:** se usa para estimar variables continuas tales como peso, temperatura, humedad, a partir de un conjunto de datos dado.
- **Regresión Logística:** se busca estimar una categoría o caracterización binaria, funciona para variables cualitativas como, alto, bajo, frío, caliente, y no requieren de una tendencia marcada.
- **Clasificador Bayesiano:** Como se mencionó anteriormente buscará atribuir una causa a partir de diferentes hechos dados, como por ejemplo clasificar si una persona es hombre o mujer, a partir de su altura, talla, peso y edad.

Cuando el conjunto de datos no incluye etiquetas que lo guíen a través de la comparación de datos para estimar un error, se acude al **aprendizaje no supervisado**, el cual busca que el algoritmo clasifique o descifre información por sí solo para clasificar o descifrar una información por su propia cuenta a partir de la observación de los datos algunos de los algoritmos más importantes del aprendizaje no supervisado son:

- Clustering K-Means
- PCA

Por último, se encuentra el **aprendizaje por refuerzo** que buscará que nuestro sistema o “agente autónomo” explore, aprenda y determine sus acciones a realizar por ejemplo los procesos de decisión markovianos.

Nota aclarativa: Durante el desarrollo del proyecto se revisarán varios de los algoritmos de aprendizaje supervisado y no supervisado de la librería Scikit-Learn en el entorno de Python el cual presenta una interfaz de implementación sencilla, por lo que para no entrar en detalle bastará con mencionar los algoritmos utilizados y como se usan.

- **Redes neuronales:** Como paralelismo de la tecnología con la naturaleza nacen las redes neuronales que busca imitar la transmisión de información que ocurre en el cerebro a través de las neuronas. El componente más sencillo de una red neuronal no es más que el **perceptrón** (o neurona artificial) el cual recibe señales de **entrada** (x) de las cuales se obtendrá una **salida** (z). A cada una de las entradas se le asignará un valor de **peso** (w) el cual definirá que tanto repercute una variable de entrada en la salida, adicionalmente cada neurona tendrá un **umbral de disparo** o “bias” (b) que activará o desactivará la neurona, dependiendo de un el valor de entrada.

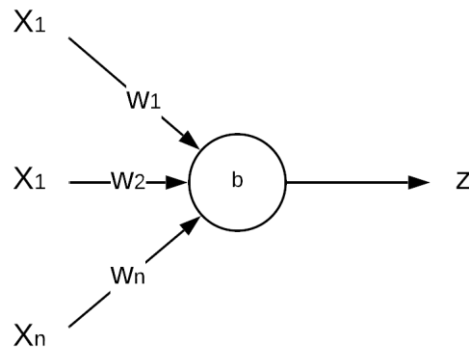


Ilustración 2. Perceptrón²

La cual se expresa de la siguiente manera:

$$z = \sum_{i=0}^n x_i * w_i - b$$

La anterior ecuación rige la salida de una neurona, las redes neuronales tienen varias formas de optimizar dependiendo si es aprendizaje supervisado o no supervisado, en el caso del aprendizaje supervisado, se requiere una fase de **entrenamiento** en la cual se proveerá de ejemplos a la red y a partir de estos se irán calibrando las variables w_1, w_2, \dots, w_n y el bias.

El proceso general para ajustar los pesos es el siguiente:

- Ajustar los pesos w_1, w_2, \dots, w_n y b de forma aleatoria.
- Aplicar las entradas x_1, x_2, \dots, x_n a la red.

² Fuente: Autor

- Calcular la salida z
- Comparar el resultado obtenido con el deseado. La diferencia entre ambos será el *error*.
- Se modifican los pesos w_1, w_2, \dots, w_n y el bias b según el *error* obtenido.
- Repetir el proceso hasta que el error esté dentro de un rango aceptable. (normalmente $< 1\%$)

Para asegurar de que la red aprenda de la mejor manera es necesario implementar un **factor de aprendizaje** λ el cual permitirá definir a qué velocidad aprende el modelo, generalmente este valor se inicializa en 0.2.

Para calcular el **error** se usará una formula sencilla:

$$e = (y - z)$$

Asumiendo que y es el valor esperado y z el valor obtenido, este error permitirá ajustar el bias sumándole el siguiente resultado:

$$\Delta b = -(\lambda * e)$$

Y al ajustar cada peso de la neurona con el siguiente delta:

$$\Delta w_i = \lambda * e * x_i$$

Hasta llegar a un nivel de error pequeño como se había mencionado, cada nuevo valor será definido como.

$$w_i^* = w_i + \Delta w_i$$

$$b_i^* = b_i + \Delta b_i$$

El modelo de perceptrón anterior es un modelo de separación lineal, pero muchas veces es necesario trabajar con problemas no lineales se deben realizar algunos cambios, de aquí nace el concepto de las redes neuronales, que no es más que la asociación de dos neuronas o más neuronas. Como ya lo se observó con el perceptrón, la capa de variables de entrada puede ser considerada como la primera capa, y la de variables de salida como la última capa, la cantidad de neuronas en cada capa estará definida por las características del problema, al agregar más capas intermedias, se habla de se hace referencia a **capas ocultas** que ayudaran a solucionar el problema de la linealización, a medida que se agregan más capas, el modelo tendrá la capacidad de adaptarse más al conjunto de los datos como se muestra en la siguiente figura.

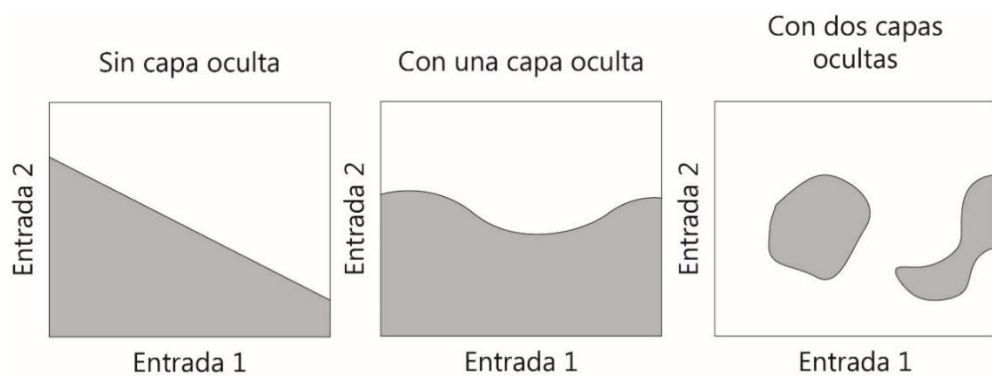


Ilustración 3. Demostración Bidimensional del efecto de las capas ocultas³

Debido a la naturaleza del algoritmo de retro propagación, el umbral de activación debe ser una función derivable por lo que generalmente se utilizan **funciones de activación** como las siguientes.

³ Fuente: Autor

- **Sigmoide:** Conocida también como función logística se expresa de la siguiente manera.

$$\varphi(z) = \frac{1}{1 + e^{-z}}$$

Dicha función es asintótica a 0 cuando toma valores muy negativos y converge a 1 cuando toma valores muy positivos.

- **Tangente hiperbólica:** Esta función de activación es similar a la función sigmoide con la diferencia notable de que esta se encuentra acotada entre -1 y 1, esta función se representa de la siguiente manera.

$$\varphi(z) = \frac{e^z - e^{-az}}{e^z + e^{-az}}$$

Estas funciones servirán para decidir una opción u otra arrojando un nivel de probabilidad para cada clase o categoría, dichas funciones deben ser aplicadas a cada neurona en las capas intermedias.

Al realizar la generalización de la acción realizada por cada neurona queda que la salida.

$$z_j^{L-p} = \begin{cases} \sum_{i=0}^{t_{L-k}} w_{ij}^{L-p} * \varphi_i^{L-p-1} - b_j & \text{si } p < L - 1, \\ \sum_{i=0}^{t_{L-k}} w_{ij} * x_i - b_j & \text{si } p = L - 1 \end{cases} \quad \forall p, k, j$$

Donde:

L es igual a la cantidad total de capas en la neurona y es quien indica en que capa se encuentra actualmente.

t es el vector topológico de la red neuronal, indica la cantidad de neuronas por capa de la siguiente manera:

$$t = \{t_1, t_2, \dots, t_i, \dots, t_L\}$$

Donde cada t_i = cantidad de neuronas en la capa i desde 1 hasta L

p es el valor de iteración entre capas desde la última capa hasta la inicial por lo que debe ir desde 0 hasta $L - 1$.

k es el valor de iteración entre capas para el índice en topología el cual debe ser contado desde la capa anterior por lo que debe ir desde 1 hasta $L - 1$.

j es el indicador de neuronas en la capa actual recorriendo todas las capas.

i es el indicador de entradas a cada neurona en la capa, el cual debería ser igual .

w_{ij}^{L-p} se leería entonces como el peso atribuido a la entrada i en la neurona j para la capa $L - p$ con $p \in \{0, 1, 2, \dots, L - 1\}$.

- **Error de una red multicapas:** Se puede considerar la función de error de varias maneras, dos de las más frecuentes son el **error medio absoluto** (MAE) y el **error cuadrático medio** (MSE) que se representan de la siguiente manera.

$$MAE(y, \varphi) = \frac{1}{n} \sum_{j=1}^{n-1} |y_j - \varphi_j| \quad MSE(y, \varphi) = \frac{1}{n} \sum_{j=1}^{n-1} (y_j - \varphi_j)^2$$

Al final se obtiene que la función de error (e) actúa con respecto al resultado de la función de activación (φ) y esta a su vez actúa en función de la suma ponderada z para todas las variables de entrada x y que varía dependiendo de los pesos w_n y el parámetro de vías b .

Se observa entonces que para **actualizar** los valores se necesitará la tasa de cambio del error con respecto a los parámetros de peso y bias, y para esto es necesario calcular el **gradiente** para cada capa, el cual se puede explicar de la siguiente manera.

$$\nabla e = \left[\frac{\partial e}{\partial w_{ij}^{L-p}}, \frac{\partial e}{\partial b_j^{L-p}} \right] \forall p \in \{1, 2, \dots, L-1\}$$

Siendo L la cantidad de capas totales, i la cantidad de variables de entrada y j la cantidad de neuronas en la capa presente. Debido a que son varias funciones las que están relacionadas se debe desglosar cada uno de los componentes de del gradiente del error a través de la **regla de la cadena**, en el caso de la última capa el valor gradiente se denota de la siguiente manera:

$$\begin{aligned} \frac{\partial e}{\partial w_{ij}^L} &= \frac{\partial e}{\partial \varphi_j^L} * \frac{\partial \varphi_j^L}{\partial z_j^L} * \frac{\partial z_j^L}{\partial w_{ij}^L} \\ \frac{\partial e}{\partial b_j^L} &= - \frac{\partial e}{\partial \varphi_j^L} * \frac{\partial \varphi_j^L}{\partial z_j^L} * \frac{\partial z_j^L}{\partial b_j^L} \end{aligned}$$

De aquí extrae el error imputado a cada neurona en la capa actual el cual será:

$$\delta_j^L = \frac{\partial e}{\partial \varphi_j^L} * \frac{\partial \varphi_j^L}{\partial z_j^L}$$

Al entender la derivada de la operación z_j de la neurona con respecto a los pesos w_{ij} y el bias b_j se obtiene que:

$$\frac{\partial z_j^L}{\partial w_{ij}^L} = \varphi_j^{L-1} \quad \frac{\partial z_j^L}{\partial b_j^L} = 1$$

Entonces los valores de actualización para los parámetros de la última capa quedan de la siguiente manera.

$$\begin{aligned} \Delta w_{ij}^L &= \lambda * \delta_j^L * \varphi_j^{L-1} \\ \Delta b_j^L &= -\delta_j^L \end{aligned}$$

Donde se observa que el cambio para los valores de peso w_{ij} en una neurona de la capa L deben estar relacionados directamente con el radio de aprendizaje λ , el error imputado a cada neurona de la capa δ_j^L y el valor de salida de la capa previa φ_j^{L-1} y el cambio en el parámetro de bias para la última capa b_j^L será el negativo del error imputado para dicha neurona. Hallando los valores para las demás capas se obtiene que los componentes de los gradientes del error con respecto a los pesos y el bias para la capa inmediatamente anterior serán expresadas de la siguiente manera.

$$\begin{aligned} \frac{\partial e}{\partial w_{ij}^{L-1}} &= \frac{\partial e}{\partial \varphi_j^L} * \frac{\partial \varphi_j^L}{\partial z_j^L} * \frac{\partial z_j^L}{\partial \varphi_j^{L-1}} * \frac{\partial \varphi_j^{L-1}}{\partial z_j^{L-1}} * \frac{\partial z_j^{L-1}}{\partial w_{ij}^{L-1}} \\ \frac{\partial e}{\partial b_j^{L-1}} &= -\frac{\partial e}{\partial \varphi_j^L} * \frac{\partial \varphi_j^L}{\partial z_j^L} * \frac{\partial z_j^L}{\partial \varphi_j^{L-1}} * \frac{\partial \varphi_j^{L-1}}{\partial z_j^{L-1}} * \frac{\partial z_j^{L-1}}{\partial b_j^{L-1}} \end{aligned}$$

Donde al tomar las ecuaciones despejadas de la última capa se obtiene lo siguiente.

$$\frac{\partial e}{\partial w_{ij}^{L-1}} = \delta_j^L * \frac{\partial z_j^L}{\partial \varphi_j^{L-1}} * \frac{\partial \varphi_j^{L-1}}{\partial z_j^{L-1}} * \varphi_j^{L-2}$$

$$\frac{\partial e}{\partial b_j^{L-1}} = -\delta_j^L * \frac{\partial z_j^L}{\partial \varphi_j^{L-1}} * \frac{\partial \varphi_j^{L-1}}{\partial z_j^{L-1}}$$

Donde $\frac{\partial z_j^L}{\partial \varphi_j^{L-1}}$ representa los pesos para la capa siguiente w_{ij}^L .

Entonces, queda que.

$$\frac{\partial e}{\partial w_{ij}^{L-1}} = \delta_j^L * w_{ij}^L * \frac{\partial \varphi_j^{L-1}}{\partial z_j^{L-1}} * \varphi_j^{L-2}$$

$$\frac{\partial e}{\partial b_j^{L-1}} = -\delta_j^L * w_{ij}^L * \frac{\partial \varphi_j^{L-1}}{\partial z_j^{L-1}}$$

El cual se puede reescribir de la siguiente manera.

$$\delta_j^{L-1} = \delta_j^L * w_{ij}^L * \frac{\partial \varphi_j^{L-1}}{\partial z_j^{L-1}}$$

$$\frac{\partial e}{\partial w_{ij}^{L-1}} = \delta_j^{L-1} * \varphi_j^{L-2}$$

$$\frac{\partial e}{\partial b_j^{L-1}} = -\delta_j^{L-1}$$

Y al generalizarla para cada una de las capas de la red neuronal se expresa de la siguiente manera.

$$\delta_j^{L-k} = \delta_j^{L-k+1} * w_{ij}^{L-k+1} * \frac{\partial \varphi_j^{L-k}}{\partial z_j^{L-k}}$$

$$\frac{\partial e}{\partial w_{ij}^{L-k}} = \delta_j^{L-k} * \varphi_j^{L-k-1}$$

$$\frac{\partial e}{\partial b_j^{L-k}} = -\delta_j^{L-k}$$

$$k \in \{1, 2, \dots, L-1\}$$

Resumiendo, se tienen las ecuaciones para actualizar los parámetros de la última red de la siguiente manera.

$$\delta_j^L = \frac{\partial e}{\partial \varphi_j^L} * \frac{\partial \varphi_j^L}{\partial z_j^L}$$

$$\Delta w_{ij}^L = \lambda * \delta_j^L * \varphi_j^{L-1}$$

$$\Delta b_j^L = -\delta_j^L$$

Donde λ es el ratio de aprendizaje definido por el usuario el cual generalmente se inicializa con un valor de 0.2, δ_j^L es el error imputado a la neurona j de la última capa y φ_j^{L-1} es la salida j de la capa anterior.

Y las ecuaciones generalizadas para las capas restantes son.

$$\delta_j^{L-k} = \delta_j^{L-k+1} * w_{ij}^{L-k+1} * \frac{\partial \varphi_j^{L-k}}{\partial z_j^{L-k}}$$

$$\Delta w_{ij}^{L-k} = \delta_j^{L-k} * \varphi_j^{L-k-1}$$

$$\Delta b_j^{L-k} = -\delta_j^{L-k}$$

$$k \in \{1, 2, \dots, L-1\}$$

Y cuando $k = L-1$

$$\varphi_j^0 = x_i$$

1.6.2. Marco Conceptual

A continuación, se definen los conceptos de los cuales se hará uso para el desarrollo del presente proyecto, con el fin de ofrecer una homogeneidad conceptual.

- **Algoritmo:** Conjunto finito de procesos a su vez finitos y bien definidos que conducen a un resultado.
- **Análisis discriminante:** El análisis discriminante es una técnica estadística que permite estudiar las diferencias entre dos o más grupos de objetos con respecto a varias variables simultáneamente. Es una técnica de clasificación y asignación de individuos a grupos. (Vallejo, 1992).
- **Aprendizaje Automático:** Conocido también como Machine Learning, Es una disciplina científica del ámbito de la Inteligencia Artificial que crea sistemas que aprenden automáticamente. Aprender en este contexto quiere decir identificar patrones complejos en millones de datos. La máquina que realmente aprende es un algoritmo que revisa los datos y es capaz de predecir comportamientos futuros. Automáticamente, también en este contexto, implica que estos sistemas se mejoran de forma autónoma con el tiempo, sin intervención humana.
- **Aprendizaje no supervisado:** Es el método de aprendizaje mediante el cual se “descubren” regularidades en los datos. Estos métodos no están dirigidos por las metas. (Moreno, y otros, 1998)
- **Aprendizaje supervisado:** Es el aprendizaje realizado a través de ejemplos los cuales deben ser *necesariamente* proporcionados para cumplir las metas de aprendizaje (Moreno, y otros, 1998)

- **Automatización:** (ST Derby, 2005, pp. 8) método de controlar automáticamente la operación de un aparato –artefacto, proceso o sistema integrado por diversos componentes a través de medios mecatrónicos– electrónicos y computacionales que sustituyen los órganos sensitivos y la capacidad de decisión del ser humano.
- **Correlación canónica:** Es una medida de asociación entre dos conjuntos de variables.
- **Cloud Computing:** Es un nuevo paradigma según el cual cualquier cosa que pueda hacerse en la informática puede trasladarse a la nube o, lo que es lo mismo, a la Red. (Fundación de la innovación Bankinter., 2011)
- **Eficiencia Técnica:** todos aquellos programas productivos que surgen de la combinación de los distintos procesos productivos de la empresa y que determinan la operativa de ésta. Por lo tanto, si se habla de mejorar la eficiencia técnica de una empresa se está haciendo referencia a la mejora de los procesos productivos y, con ello, a la eficiencia operativa global de la empresa.
- **Globalización:** La globalización es un proceso económico tecnológico político, social, empresarial y cultural a escala mundial que consiste en la creciente comunicación e interdependencia entre los distintos países del mundo uniendo sus mercados, sociedades y culturas, a través de una serie de transformaciones sociales, económicas y políticas que les dan un carácter global.
- **Inteligencia Artificial:** Nebendahl [1988] y Delgado [1998], la definen como el campo de estudio que se enfoca en la explicación y emulación de la conducta inteligente en función de procesos computacionales basadas en la experiencia y el conocimiento continuo del ambiente.

- **Internet de las cosas:** Es un concepto que consiste en que tanto personas como objetos puedan conectarse a internet en cualquier momento y lugar. Es decir, los elementos de uso cotidiano vienen con aplicativos para acceder a internet y conectarse con otros elementos solamente con estar dentro de la misma red o incluso al otro lado del mundo. (Fundación de la innovación Bankinter., 2011)
- **Perceptrón:** Es la neurona artificial o unidad básica de inferencia en forma de discriminador lineal, a partir del cual se desarrolla un algoritmo capaz de generar un criterio de para seleccionar un subgrupo a partir de un grupo de componentes más grande. (Rosenblatt, 1958)
- **Redes Neuronales:** Existen numerosas formas de definir una red neuronal, pero preferimos quedarnos con la definición más acertada:
- *“Una red neuronal es un sistema de computación compuesto por un gran número de elementos simples, interconectados (perceptrón), los cuales procesan información por medio de su estado dinámico como respuesta a entradas externas”.* (Matich, 2001)
- **Regresión:** Consiste en la relación matemática determinística más simple entre dos variables X y Y siendo una relación lineal $y = \beta_0 + \beta_1 x$. El conjunto de pares (x, y) para los cuales $y = \beta_0 + \beta_1 x$ determina una línea recta con pendiente 1 e intersección en y = 0.
- **Sistema Ciber-Físico:** integra capacidades de computación, almacenamiento y comunicación junto con capacidades de seguimiento y/o control de objetos en el mundo físico. Los sistemas ciber físicos están, normalmente, conectados entre sí y a su vez conectados con el mundo virtual de las redes digitales globales

- **Virtualización:** método de dividir un servidor físico en múltiples servidores ficticios o “virtuales”, dando a cada uno el aspecto y la capacidad de estar funcionando en su propia máquina dedicada. Cada servidor virtual funciona como un servidor de pleno derecho y puede ser reiniciado de forma independiente.

1.6.3. Marco Temporal

Este proyecto se enmarca en momentos donde existe un creciente interés por las Tecnologías de la información y la comunicación (TIC) aplicada en múltiples sectores, uno de ellos el sector empresarial el cual presenta una necesidad de desarrollo a través de la optimización de procesos y la aplicación de herramientas que permitan automatizar y mejorar la calidad de procesos claves como la medición y control de variables.

Siendo consecuentes con la oleada de la industria 4.0 y las tendencias vistas alrededor del mundo y en la región entorno a las tecnologías de Información y, adicional a esto, considerando como última instancia, la culminación del ciclo formación profesional, se plantea el desarrollo del proceso investigativo, en el marco de la duración de las actividades impartidas desde el anterior semestre académico 2019-2 hasta el semestre actual en curso, es decir con 2020-1, sujetos a los cambios de la directiva de la Universidad Tecnológica de Pereira y con relación al calendario académico vigente, con el fin de completar dichas actividades junto con la culminación del actual semestre.

1.6.4. Marco Espacial

El trabajo se desarrolla en la Universidad tecnológica de Pereira en los laboratorios de Facultad de ciencias empresariales, el cual es un entorno compuesto por tres laboratorios: Laboratorio de GEIO enfocado en la investigación y enseñanza de temáticas pertenecientes a la Ingeniería industrial impartiendo el uso de metodologías lúdicas para el aprendizaje de conceptos trabajados en el marco de la ingeniería industrial y las ciencias empresariales, el laboratorio móvil de Logística enfocado en la investigación y divulgación de buenas prácticas de almacenamiento valiéndose de metodologías experimentales para estudiar herramientas y modelos logísticos en el interior de las organizaciones y por último el laboratorio de manufactura Flexible en el cual se realizan prácticas de aplicación de herramientas tecnológicas a la simulación de procesos de manufactura.

1.7.METODOLOGÍA DE LA INVESTIGACIÓN

1.7.1. Diseño Del Estudio

- **Exploratorio:** El estudio exploratorio se desarrolla con el propósito de visualizar, exponer y definir los diferentes modelos existentes en el marco de la inteligencia artificial o más en profundidad en el campo del machine learning que estén o puedan ser aplicados a diferentes tipos de procesos industriales. para esto se realizará una búsqueda documental y bibliográfica de los diferentes modelos y sus indicadores para posteriormente ser enlistados y clasificados para facilitar la comparativa entre estos modelos.

- **Descriptivo:** se considera como investigación descriptiva aquella en que, como afirma salkind (1998), "se reseñan las características o rasgos de la situación o fenómeno objeto de estudio" (p.11).según cerda (1998),"tradicionalmente se define la palabra describir cómo el acto de representar ,reproducir o figurar a personas ,animales o cosas ..." y agrega: "se deben describir aquellos aspectos más característicos ,distintivos y particulares de estas personas ,situaciones o cosas ,o sea, aquellas propiedades que las hacen reconocibles a los ojos de los demás". Con el desarrollo de la metodología descriptiva, será posible definir el problema de estudio de los modelos de automatización y electroneumática de Fischertechnik con lo cual se podrá extraer posteriormente la interacción de sus variables y poder así simularlas. A partir del proceso de simulación, se pretenderá realizar la implementación de cada uno de los modelos definidos en la fase exploratoria y posteriormente poder discriminar los modelos con mejor ajuste o aplicación de acuerdo con los criterios identificados para cada modelo.
- **Aplicado:** Se pretende observar la aplicación de diferentes modelos de aprendizaje automático a un aspecto real observado en los módulos de FischerTechnik del laboratorio perteneciente a la facultad de ciencias empresariales, por lo que posterior a la aplicación de los algoritmos, se usarán los datos arrojados por dichos modelos para alimentar los módulos, y así observar el impacto real de la implementación de dichas herramientas.

La investigación a realizar deberá ser de carácter mixto teniendo en cuenta el tipo del estudio exploratorio y descriptivo, ya que se pretende identificar los tipos de modelos de aprendizaje automático existentes (lo cual corresponde a la fase exploratoria) y posteriormente comprobar su aplicación en un entorno controlado como lo es el proceso de simulación de los módulos de FischerTechnik para esto será necesaria la

aplicación de una fase descriptiva para poder extraer las diferentes variables del proceso y sus relaciones y posteriormente interpretando sus variables en el mundo real. los modelos de aprendizaje automático identificados y los modelos descritos darán pie a un tipo de estudio experimental ya que se realizará el análisis de relación entre las variables tomadas del proceso alimentadas por la simulación y su interacción con los modelos de aprendizaje automático, mediante el cambio de herramientas, se pretende determinar cuáles son aplicables, y de ser posible, detectar qué modelos de aprendizaje influyen positivamente en el caso de estudio productivo.

1.7.2. Metodología

Se desea corroborar una hipótesis planteada acerca de la aplicación de diferentes herramientas a un contexto específico, implementando herramientas de aprendizaje automático y conceptos de simulación a un problema tangible en los módulos de automatización y electroneumática de Fischertechnik. Para la actual investigación, la metodología a seguir será *Hipotético-deductivo* según Bernal (2016), el método hipotético - Deductivo “consiste en un procedimiento que parte de unas aseveraciones en calidad de hipótesis y busca refutar o falsear tales hipótesis, deduciendo de ellas conclusiones que deben confrontarse con los hechos”. En otras palabras, se espera que a través de la descripción, explicación, aplicación y verificación de diferentes modelos de aprendizaje automático a los módulos de Fischertechnik se acepte o rechace la hipótesis de que estas herramientas son aplicables y favorecen el desempeño de los procesos descritos a través de la experimentación, obteniendo datos a través de la experimentación y la simulación. (Rodríguez, & Pérez; 2017).

Para la selección de la técnica y fuentes de información se tiene que la información necesaria será de carácter teórico por ende no se utilizarán fuentes primarias, sino que

se harán uso de fuentes secundarias como diferentes libros de texto, documentos, revistas, blogs digitales, entre otros, con el fin de encontrar los diferentes modelos relacionados al tema y resultados de posibles aplicaciones previas.

1.7.3. Fuentes de información:

Para la recolección de los datos, se realizará uso de fuentes secundarias de datos a modo de *Análisis Documental*. Las diferentes fuentes de información que servirán de base para la investigación se componen, principalmente de libros de texto enfocados al marco de machine learning como lo son:

Inteligencia Artificial. Fundamentos, Práctica y Aplicaciones, En el cual se encuentran registrados algunos de los ejemplos básicos de aplicación de la inteligencia artificial, además de que relaciona algoritmos vistos durante la carrera y expone la investigación de operaciones como uno de los precursores de la inteligencia artificial. El libro escrito por García Serrano (2017) permitirá exponer los fundamentos teóricos y prácticos detrás de la inteligencia artificial, y la entrada al machine learning a través de algoritmos de búsqueda tanto informada como no informada.

Big Data con Python: Este libro presente conocimientos y tecnologías relacionadas con el análisis de datos a gran volumen, basado en el entorno de programación de Python, y pasando por diferentes herramientas como MongoDB y Spark.

repositorios, Blogs y Guías de Usuario de los diferentes programas como lo son:

OPEN AI: Es una compañía de investigación de inteligencia artificial sin fines de lucro. Su objetivo es avanzar en la inteligencia digital de la manera que es más probable que beneficie a la humanidad en su conjunto, sin restricciones por la necesidad de generar un retorno financiero. Se cree que la IA debería ser una extensión de la voluntad humana individual y, en espíritu de libertad, distribuida de la manera más amplia y uniforme posible.

AI. GOOGLE: La misión de Google es organizar la información del mundo y hacerla universalmente accesible y útil. Es un soporte para el estado del arte en el campo de la Inteligencia Artificial, aplicando dicha herramienta a productos y nuevos dominios, y desarrollando herramientas para garantizar que todos puedan acceder a ella.

Otras posibles fuentes de datos son los modelos simulados de los módulos del FischerTechnik, de los cuales se podrán extraer buena cantidad de datos del proceso. Se usarán modelos de simulación para obtener estimaciones de características de operación en mucho menos tiempo del que se requiere para recopilar esos mismos datos de operación a partir del sistema real de los visualizado en los módulos de Fischertechnik.

CAPITULO 2. OBTENCIÓN DE DATOS

2.1.FISCHERTECHNIK (CONTEXTUALIZACIÓN)

Fischertechnik es una empresa alemana enfocada a la producción de kits de ensamble los cuales constan de distintos tipos de piezas de carácter mecánico, electroneumático y electrónico digital, proporcionando a todo tipo de estudiantes y profesores las herramientas suficientes para la implementación en sus aulas de clases basados principalmente en la metodología STEM.

Para el desarrollo del presente proyecto se hace uso del set de construcción “STEM Engineering de Fischertechnik”⁴ el cual permite desarrollar 22 modelos distintos dentro de los módulos de Industria, Exploración y Electroneumática. De los posibles proyectos que ofrece el kit de ensamble se decide iniciar la construcción del modelo *Robot Clasificador por colores*, debido a que este cuenta con sensores de un grado más alto de complejidad comparado con otros modelos del kit de ensamble.

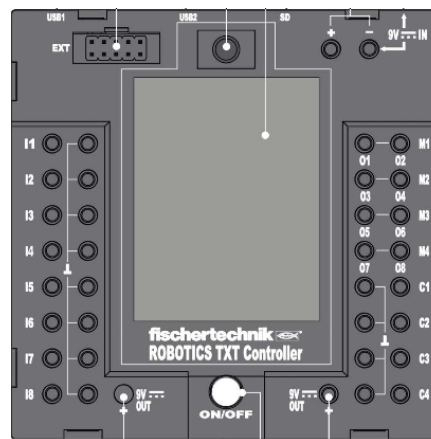
2.2.ELEMENTOS

A continuación, se describirán los elementos electrónicos y electroneumáticos que permitirán desarrollar la construcción del Robot Clasificador por Colores, dentro de la clasificación de los elementos se cuenta tres Grupos, los Actuadores, los Sensores y El controlador los cuales componen los elementos básicos de un autómata programable.

⁴ Ver información en: <https://www.fischertechnik.de/es-es/productos/aprender/stem-sets/519341-stem-engineering>

2.2.1. Controlador

El controlador Lógico Programable o PLC, es el canal de comunicación entre las entradas y las salidas del modelo, las cuales son gestionadas por los sensores y actuadores respectivamente, en este se realiza la programación del proceso, creando, almacenando y ejecutando ordenes relacionadas con datos de entrada.



*Ilustración 4. TXT Controller*⁵

El controlador TXT de FischerTechnik cuenta, entre otras cosas, con hembra de conexión tipo USB, entradas universales I1-I8, contadores C1-C4, Salidas M1-M4 o O1-O8 como puertos de conexión de los elementos presentes en el mundo real y sus interacciones dentro del proceso que se programan de forma Digital. Para la programación del modelo en cuestión se requerirá del uso del software RoboPro versión 4.0 o superior y una alimentación de 9 V. Otra herramienta que permitirá realizar la programación del controlador será el Framework *ftrobopy* que permite integrar herramientas de FischerTechnik con otros paquetes de trabajo dentro del lenguaje de programación Python.

⁵ Fuente: https://content.ugfischer.com/cbfiles/fischer/Zulassungen/ft/TXT-Controller_es.pdf

2.2.2. Actuadores

Los actuadores proporcionan los medios para que el modelo actúe en el entorno. Son las salidas del dispositivo: los motores, las luces, los compresores etc. Estos elementos convierten energía eléctrica en energía mecánica, produciendo movimientos en el mundo real. Asimismo, es un dispositivo de salida que recibe una orden del controlador y ambos efectúan acciones para modificar el estado de un equipo (calefacción, iluminación, sonido, etc.). En ciertas ocasiones un actuador puede estar integrado con un sensor. Este dispositivo se encarga de ejecutar la acción requerida por el controlador y así proceder a realizarla.

Entre los actuadores usados para el modelo del clasificador por color se encuentran los siguientes:

- **Válvula Solenoide de 3/2 vías.**

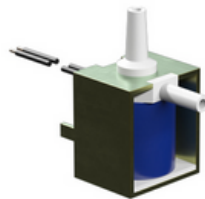


Ilustración 5. Válvula Solenoide⁶

Acción: Regular el flujo de aire.

Energía que provee: Neumática.

Descripción: Son válvulas que están controladas por un servomotor. Las más comunes son las válvulas todo-nada, que cuando actúa el motor abre o cierra el

⁶ Fuente: <https://www.fischertechnik.de/es-es/productos/piezas/neumatica/35327-valvula-solenoide-de-3-2-vias-9-12v>

circuito. Con las válvulas solenoides se puede optimizar el funcionamiento correcto del circuito, ya que van a actuar en función de impulsos que estarán regulados por relés. (Ostos, P. (2012))

Variables que afectan el desempeño: Fugas en los circuitos neumáticos.

- **Mini motor.**



Ilustración 6. Minimotor de 6-9V⁷

Acción: Generar una Fuerza par constante.

Energía que provee: Mecánica

Descripción: Un motor, es un receptor que, al ser alimentado mediante una corriente eléctrica, produce un movimiento giratorio en su eje que, a través de los acoplamientos mecánicos adecuados, es aprovechado para efectuar diferentes trabajos en el sector industrial y doméstico. Generalmente el funcionamiento y arranque de los motores eléctricos suele estar gestionado por sistemas de automatismos eléctricos. (Martín, J, García, M. (2016)).

Variables que afectan desempeño: Alimentación intermitente

⁷ Fuente: <https://www.fischertechnik.de/es-es/productos/piezas/electronica/32293-mini-motor-6-9v-0-65-a-max-negro>

- **Compresor**

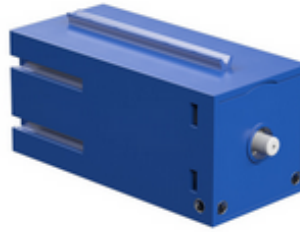


Ilustración 7. Compresor⁸

Acción: Generar un constante flujo de aire comprimido.

Energía que provee: Neumática

Descripción: Los compresores son aparatos o maquinas mediante las cuales se obtiene aire a presión de forma continuada y con las menores perturbaciones posibles, de forma que aseguren la uniformidad en la presión de la red y una calidad apropiada de aire neumático en pureza y humedad para el tipo de utilización que tenga, Existen compresores alternativos y compresores rotativos. (Roldán, J. (2012))

Variables que afectan desempeño: Fugas en los conductos.

- **Bombillo**



Ilustración 8. Bombilla Insertable⁹

Acción: Generar una señal lumínica.

⁸ Fuente: <https://www.fischertechnik.de/es-es/productos/piezas/neumatica/121470-compresor-9v>

⁹ Fuente: <https://www.fischertechnik.de/es-es/productos/piezas/electronica/37875-foco-bombilla-insertable-con-lente-9v-0-15a>

Energía que provee: Lumínica

Descripción: El funcionamiento de esta lámpara se basa en que un filamento de tungsteno de espiral simple o doble se lleva hasta la incandescencia con el paso de la corriente eléctrica, con el objeto de que no se queme el filamento, se encierra en una ampolla o bulbo de vidrio dentro del cual se hace vacío o se introduce un gas inerte. Se suele utilizar en conjunto con fototransistores para la detección de presencia-ausencia en aplicaciones industriales. (Enríquez, G. (2004).)

Variables que afectan desempeño: Alimentación intermitente

2.2.3. Sensores

Un sensor es un dispositivo que mide fenómenos del mundo físico. Puede detectar magnitudes físicas o químicas, variables de instrumentación y transformarlas en variables eléctricas. Estos fenómenos pueden ser cosas tales como luz, humo, vapor de agua, humedad, fuerza, torsión, movimiento etc. Los sensores pueden ser analógicos o digitales y normalmente se diseñan para medir sólo una cosa o concepto. Los sensores pueden ser activos o pasivos. Los sensores activos inyectan energía en el entorno (medio ambiente) para detectar cambios de algún tipo. Los sensores pasivos detectan energía que ya está allí.

- **Pulsador**

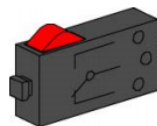


Ilustración 9. Pulsador.¹⁰

¹⁰ Fuente: <https://www.fischertechnik.de/es-es/productos/piezas/sensores/37783-mini-interruptor-sensor-de-contacto-rt-negro>

Tipo de variable a medir: Presencia/ausencia

Tipo de dato: Digital (1 - 0)

Descripción: los interruptores o pulsadores cuentan como sensor de contacto el cual al activar el botón rojo variará su estado de acuerdo con la configuración con la que se haya conectado previamente.

Su señal es digital y por lo que marcará entre 0 y 1, éstos pueden ser usados como “normalmente abiertos” o “normalmente cerrados”.

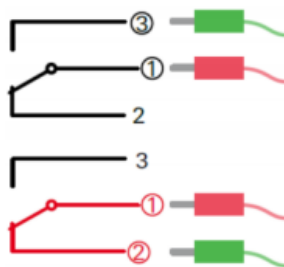


Ilustración 10. Conexiones del Pulsador.¹¹

En la imagen anterior se observa el modo de conexión de los contactos para el sensor, con los contactos 1 y 3 el sensor se encuentra “normalmente abierto”, mientras que con los contactos 1 y 2 el sensor se encuentra “normalmente cerrado”.

- **Foto transistor**



Ilustración 11. Foto Transistor.¹²

¹¹ Fuente: <https://content.ugfischer.com/cbfiles/fischer/Zulassungen/ft/37783-Mini-switch.pdf>

¹² Fuente: <https://www.fischertechnik.de/es-es/productos/piezas/electronica/36134-fototransistor-insertado-en-enchufe-amarillo>

Tipo de variable a medir: Presencia/ausencia

Tipo de dato: Digital (1 - 0)

Descripción: el fototransistor o también llamado como “Sensor de luminosidad”, es un sensor que reacciona ante la claridad o radiación lumínica. En el caso de una barrera de luz, es la contracara para la lámpara que emite el haz de luz, cuando el transistor se encuentra iluminado, este transmite una corriente que va desde los 15 mA hasta los 75 mA en corriente de alta.

Debe contar con un actuador generador de luz como un bombillo led o una bombilla incandescente, los cuales, juntos, funcionan como una barrera lumínica que detecta presencia o ausencia de un objeto de acuerdo con la programación establecida con anterioridad.

Variables que afectan la medición: Condiciones Lumínicas

- **Sensor de Color**

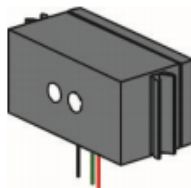


Ilustración 12. Sensor de Color¹³

Tipo de variable a medir: Color (RGB)

Tipo de dato: Analógico (0 – 2000 mV)

Descripción: el sensor de color de Fischertechnik cuenta con una fuente de luz roja que se refleja en distinto nivel de intensidad según la superficie de color que se

¹³ Fuente: <https://content.ugfischer.com/cbfiles/fischer/Zulassungen/ft/128599-Color-sensor.pdf>

encuentre en frente de este, la intensidad de luz reflejada se mide mediante un fototransistor y se detecta como un valor analógico entre 0 y 2000 mV.

Variables que afectan la medición: El valor de medición depende tanto de la luminosidad ambiente como de la distancia del sensor a la superficie de color.

Los valores del sensor de color se clasificarán para blanco, rojo y azul según el valor asignado por el sensor, debido a que los valores de asignación para cada color son fuertemente dependientes de las condiciones físicas de la medición deberá asegurarse una correcta calibración para los rangos definidos para cada color.

2.2.4. Robot clasificador por colores

Teniendo en cuenta los componentes dentro del set de Engineering de Fischertechnik disponible en los laboratorios de la facultad de ciencias empresariales de la Universidad Tecnológica de Pereira, se construye el modelo llamado “*Robot Clasificador por colores*” siguiendo los pasos mostrados en el *Anexo A*. El robot Clasificador por colores sirve para separar de forma automatizada elementos de distintos colores, los objetos de igual geometría pero de distinto color se conducen frente al sensor por medio de un brazo mecánico el cual tiene una ventosa en el extremo que le permite sujetar los objetos mientras un motor de giro le permite al brazo desplazarse a la ubicación contigua, el recorrido que realiza el brazo se realiza con un contador de pulsos, posterior a la selección de color, el brazo se encargará de llevar cada objeto a su respectivo lugar de almacenamiento, según el color que el sensor le haya asignado.

El modelo cuenta con componentes tanto mecánicos como electroneumáticos los cuales, para términos prácticos se mostrarán por aparte, aunque la instalación de estos componentes a lo largo de la construcción del modelo sea relativamente paralela.

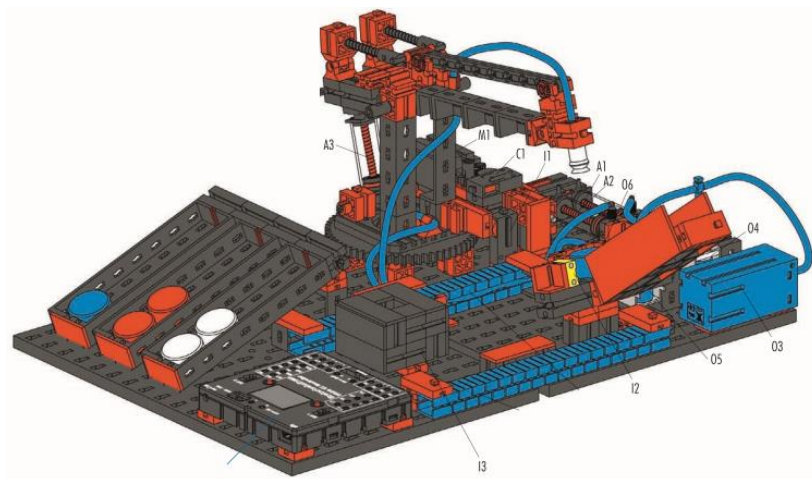


Ilustración 13. Robot Clasificador por color.¹⁴

Dentro de la parte mecánica se encuentran todos los componentes físicos tanto móviles como fijos que sirven de infraestructura en la cual se sustenta el proceso realizado por el modelo como se muestra en la figura anterior, mientras que, el eje de electropneumática sirve de soporte a los movimientos a través de los anteriormente mencionados cilindros, los cuales son alimentados por el flujo producido a partir del compresor.

¹⁴ Fuente: <https://www.fischertechnik.de/es-es/productos/aprender/stem-robotics/533019-robotics-electropneumatics>

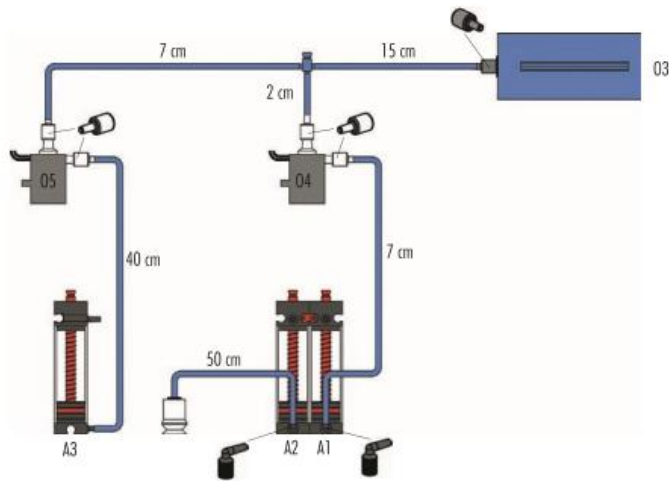


Ilustración 14. Circuito Neumático.¹⁵

2.2.5. Asignación De Entradas Y Salidas

- **Entradas universales I1–I8:** Son los genios universales entre las entradas de señal. Son configurables a través del software ROBO Pro para:

Sensores digitales (pulsadores, relés de láminas, fototransistores); digital de 5 k Ω .

Sensores infrarrojos de pista; digital de 10 V.

Sensores analógicos de 0–5 k Ω (resistores NTC, fotorresistores, potenciómetros)

Sensores analógicos de 0–10 V (sensores de color), indicación del valor en mV (milivoltios.)

¹⁵ Fuente: Anexo A. Construcción Robo Clasificador por Colores.pdf

Tabla 1. Entradas universales I1-I8

| Tipo de sensor | Descripción | Numeración | Notación |
|-----------------------|--|-------------------|-----------------|
| Pulsador | Pulsador de paro para posición inicial | Sensor 1 | I1 |
| Foto transistor | Sensor de Presencia de producto | Sensor 2 | I2 |
| Sensor de Color | Lector de Intensidad lumínica reflejada por los objetos. | Sensor 3 | I3 |

Entradas C1–C4

Entradas rápidas de conteo que registran impulsos de conteo hasta 1 kHz (1000 impulsos/segundo), p. ej. de los motores de codificador FischerTechnik. También utilizables como entradas digitales, p. ej. para pulsadores.

Tabla 2. Entradas de contador

| Tipo de sensor | Descripción | Numeración | Notación |
|-----------------------|---------------------------------------|-------------------|-----------------|
| Pulsador | Pulsador de registro de giro de brazo | Sensor 1 | C1 |

Salidas M1–M4 o O1–O8

En las salidas se pueden conectar 4 motores. Como alternativa 8 lámparas o electroimanes, cuyo segundo polo se conecta a una conexión de masa (L).

Tabla 3. Salidas del Modelo.

| Tipo de Motor | Descripción | Numeración | Notación | |
|-----------------------------|---|-------------------|-----------------|----|
| Mini motor | Giro dirección 1 | Actuador 1 | O1 | M1 |
| | Giro dirección 2 | Actuador 1 | O2 | |
| Compresor | Encender compresor | Actuador 2 | O3 | M2 |
| Electroválvula solenoide de | Apertura de Compuerta (presurización) - Ventosa. | Actuador 3 | O4 | |
| Electroválvula solenoide de | Apertura de Compuerta (presurización) - Elevación y descenso. | Actuador 4 | O5 | M3 |
| Bombilla | Generar haz de luz para el fototransistor. (presencia/ausencia de materia prima | Actuador 5 | O6 | |

A continuación, se observan las conexiones de tanto las entradas como las salidas del modelo al controlador TXT .

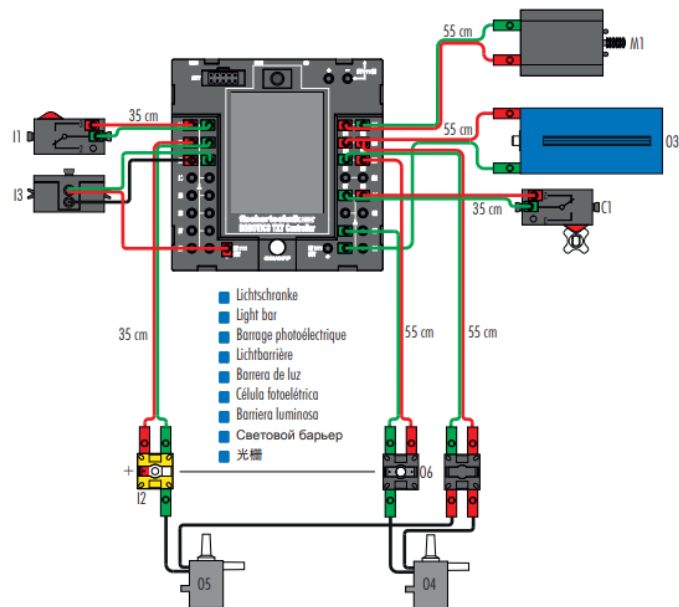


Ilustración 15. Conexión TXT Controller para robot clasificador por color¹⁶

2.3.RUTINA DEL ROBOT CLASIFICADOR DE COLOR

El proceso de selección por color depende de los siguientes subprocesos:

1. Posición Inicial: Para iniciar el proceso debe asegurarse que el brazo se encuentre en su posición inicial o punto de partida, para esto se hará uso del sensor I1, si este se encuentra inactivo, significa que el brazo no se encuentra en la posición inicial y debe activarse el motor 1 en polaridad negativa (M1-) para ir a la posición inicial hasta que el sensor I1 se active, de otra forma, si el brazo se encuentra en su posición inicial continua al siguiente subproceso.
2. Verificación de materia prima: Verificar la existencia de materia prima en la entrada del proceso, para esto es necesario activar primero la salida O6 y posteriormente con

¹⁶ Fuente: <https://content.ugfischer.com/cbfiles/fischer/Zulassungen/ft/Zusatzblatt-TXT-Schaltplaene-fuer-ElectroPneumatic.pdf>

el fototransistor I2 verificar la presencia de material en la entrada, si se detecta presencia de materia prima, pasará al siguiente subproceso, de lo contrario no se realizará ninguna acción.

3. Recoger Materia Prima: Se inicia la alimentación del flujo neumático mediante la activación del compresor alojado en la salida O3 y posteriormente se acciona el brazo mecánico mediante el solenoide conectado al O5, al accionar el actuador, el brazo descenderá sobre la materia prima y posteriormente activar la ventosa conectada al terminal de salida O4, para subir el brazo mecánico es necesario cortar el flujo de corriente en el actuador en la salida O5, debido a que el cilindro alimentado por este solenoide es un cilindro con muelle, este volverá a la posición inicial antes de activar dicho solenoide.
4. Llevar a sensor de color: Para transportar la materia prima al sensor de color es necesario activar el Motor 1 con polaridad positiva (M1+), cuando el contador de pasos C1 incremente en 6 pulsos es necesario detener el motor 1 (M1°) y posteriormente activar el flujo de corriente en la salida O5 para alimentar el cilindro A3 y hacer que el brazo descienda para poder seguir al siguiente subproceso.
5. Tomar valor para el color de la materia prima: El reconocimiento del color lo realiza un sensor óptico que, a base de la reflexión de una superficie, deduce su color, así pues, el sensor de color es un sensor de reflexión que indica que tanto refleja una superficie la luz roja. Esto significa que la medición del sensor no es proporcional a la longitud de onda del color medido y no puede ser trabajado en coordenadas de color (como por ejemplo RGB o CYMK). La diferenciación de las piezas de color se realizará entonces mediante valores de umbral (los cuales serán denominadas como α y β) que delimitan los valores de cada color. Dado que los valores de umbral para cada sensor de color suelen ser distintos, debe asegurarse un proceso previo de calibración de estos límites de umbral los cuales se usan para discriminar los colores de la siguiente manera:

$$I3 < \alpha, \text{ Blanco}$$

$\alpha \leq I3 < \beta$, Rojo

$\beta \leq I3$, Azul

Donde para el software RoboPro cuenta con los valores por defecto de (1590 y 1800) respectivamente y el Framework ftrobopy inicializa sus valores en (200 y 1000) respectivamente.

6. Definir lugar de almacenamiento según color: según el valor arrojado por el sensor posterior a ser calibrados sus valores de umbral se determinará la cantidad de pulsos necesarios para que el brazo pueda llegar a cada uno de los tres compartimientos de almacenamiento determinados de la siguiente manera:

Almacén Blanco = 4 Pulsos en C1

Almacén Rojo = 6 Pulsos en C1

Almacén Azul = 8 Pulsos en C1

Así pues, posterior a la determinación de los pulsos necesarios para el almacenamiento se realizan tres bifurcaciones para llevar la materia prima a su respectivo comportamiento como se observa en el siguiente subproceso.

7. Llevar a almacén: las tres bifurcaciones en la programación realizan exactamente las mismas acciones, exceptuando por el conteo de los pasos que determinan cuando detener el motor.

Una vez determinado el color se corta el flujo de corriente de la salida O5 lo cual le permitirá al brazo subir, se activa el motor M1 en polaridad positiva (M1+) para mover el brazo hacia los compartimientos y se inicia con el conteo, dependiendo de la decisión del sensor para sus umbrales de color, se asigna una de las tres cantidades de pulsos para el almacén, posterior a contar la cantidad exacta de pulsos, se detiene el motor 1 (M1°) y se activa el flujo de energía al solenoide O5 el cual a su vez permite el flujo de aire al cilindro A1, una vez el brazo haya descendido se limita el flujo en el solenoide O4 para apagar la ventosa y que el material pueda caer en su ubicación.

8. Volver a posición Inicial: Por último, resta volver a la posición inicial y para esto se limitará el flujo del solenoide O5, es ahí cuando el brazo sube y se procede a activar el Motor 1 en polaridad Negativa (M1-) el realizará el recorrido hasta que el sensor I1 se encuentre activo, y allí cortar el flujo al Motor 1 (M1°).

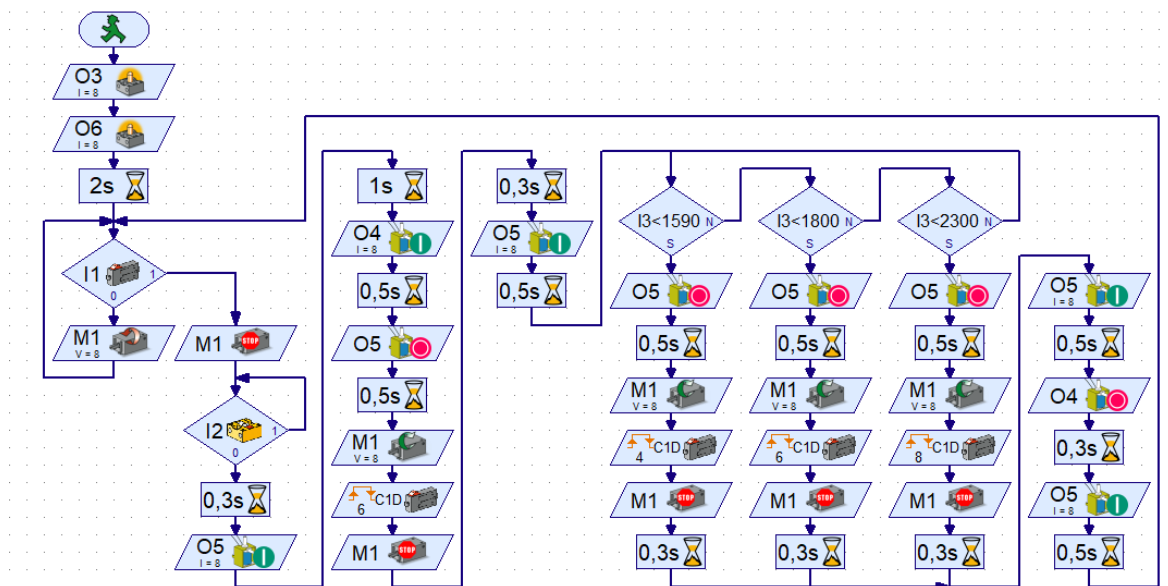


Ilustración 16. Programación en RoboPro.¹⁷

2.4. OBTENCIÓN DE LOS DATOS

Es importante destacar que para la implementación de los distintos algoritmos de machine learning es necesario contar con una gran cantidad de datos organizados y a su vez contar con el entendimiento de estos. Debido a la situación que se está viviendo actualmente en época de pandemia, se dificulta el acceso al laboratorio para realizar mediciones físicas además de que sería ineficiente realizar todas las mediciones manualmente por lo que se decide implementar sistemas de simulación del proceso, el cual permitirá no solamente

¹⁷ Fuente: Autor

visualizar el funcionamiento del modelo de una manera virtual, sino que también permitirá generar un conjunto de datos más completo y en menos tiempo.

Para la simulación del proceso se realizarán algunos supuestos en lo que a la clasificación del color respecta, de acuerdo con las mediciones realizadas previamente para cada uno de los colores se generan tres distribuciones de probabilidad y con estas realizar el generador de procesos para cada color.

Asumiendo que la cantidad de radiación reflejada por cada color varía con un comportamiento normal oscilando su valor medido en el sensor alrededor de 565 mV para el blanco, 833 mV para el rojo y 1500 mV para el azul, se usa una desviación de 135 la cual permite visualizar algunas particularidades entre las distribuciones y favorecerá el análisis para el calibrador.

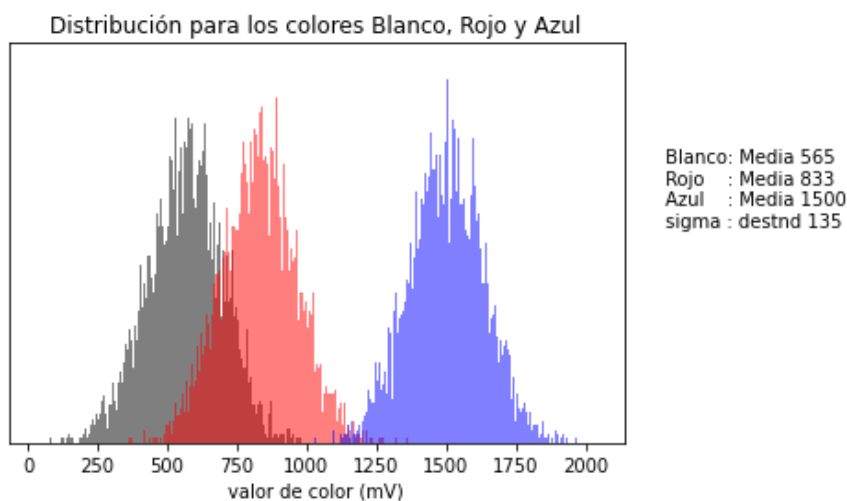


Ilustración 17. Distribución por color.¹⁸

¹⁸ Fuente: Autor

Para calibrar el sensor de color se aplicará un algoritmo de aprendizaje que se encargará de estimar el error en la medición y actualizará los valores de α y β hasta encontrar un punto estable.

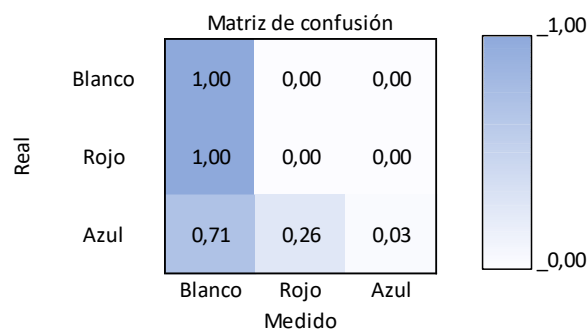


Ilustración 18. Matriz de confusión 1.¹⁹

En la imagen anterior se observan los valores de la matriz de confusión inicializando los límites de umbral α y β en 1590 y 1800 mV respectivamente mientras que en la siguiente imagen se obtiene la matriz de confusión para la clasificación con valores en 200 y 1000 mV para los límites de umbral.

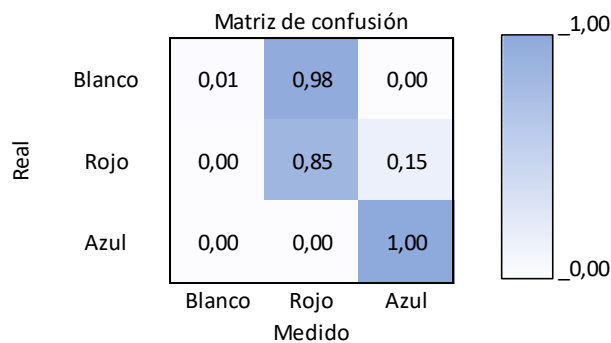


Ilustración 19. Matriz de Confusión 2.²⁰

¹⁹ Fuente: Autor.

²⁰ Fuente: Autor

Se observa que tanto el valor de inicialización asignado en la programación a través de RoboPro ($\alpha = 1590$, $\beta = 1800$) y el valor asignado por defecto en la librería ftrobopy ($\alpha = 200$, $\beta = 1000$) son malos valores de inicialización pues el clasificador ignora un color a favor del otro, esto debido a que para los valores asignados en la distribución de Media y desviación para el color blanco y el color rojo, hacen que estos se solapen entre sí en un amplio espectro, yendo desde aproximadamente los 500 mV hasta los 1000 mV.

2.5.CALIBRADOR

Para calibrar el sensor de color se utiliza el método de reducción del error el cual se encuentra documentado en el Anexo B.

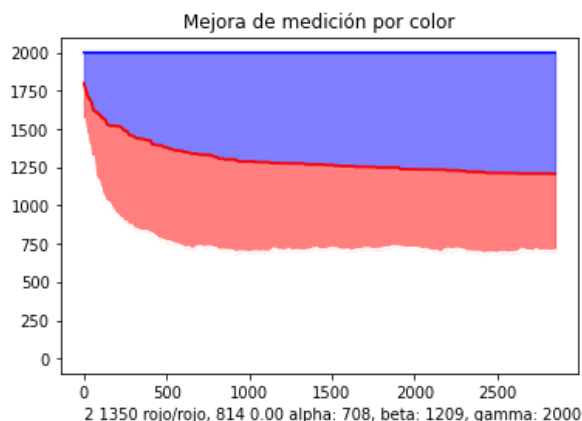


Ilustración 20. límites de umbral calibrados desde valores altos.²¹

En la figura anterior se observa que cuando se inicializan los valores de α y β en 1590 y 1800 mV éstos valores de umbral tienden a estabilizarse alrededor de 700 y 1200 mV respectivamente.

²¹ Fuente: Autor

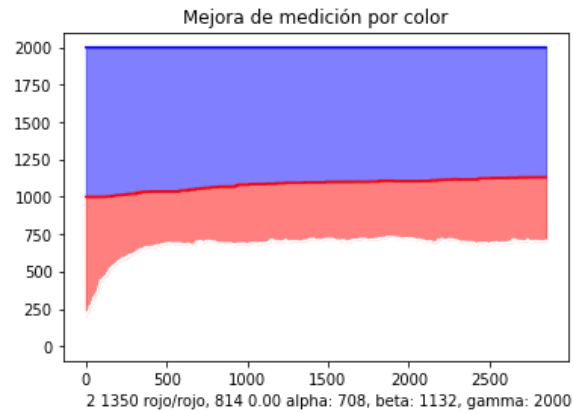


Ilustración 21. Límites de umbral calibrados desde valores bajos.²²

Mientras que para los valores de inicialización de 200 y 1000, si bien el parámetro de umbral α tiende a estabilizarse igualmente en un valor cercano a los 700 mV, el valor de umbral β tiende a alejarse más de la medición.

Al realizar una mayor cantidad de repeticiones los valores tienden a estabilizarse alrededor de los 720 mV y 1120 mV para α y β respectivamente por lo que se tomarán estos valores de umbral como la calibración óptima del sensor, y se asignarán dentro del proceso para la generación de datos y su posterior análisis.

²² Fuente: Autor

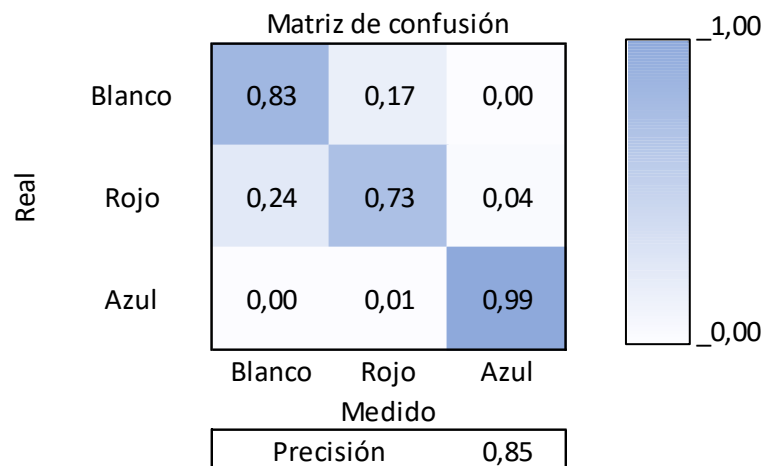


Ilustración 22. Matriz de confusión para los valores con límites de umbral calibrados.²³

Los valores de precisión para cada color con la nueva calibración de los límites de error para los datos de entrenamiento se muestran en la matriz anterior y se observa una leve tendencia a confundir el color rojo con el blanco, no obstante, el proceso global de medición se encuentra calibrado con 85% de precisión, suficiente para continuar con la generación de nuevos datos y alimentar los modelos para su testeo.

²³ Fuente: Autor

2.6.GENERACIÓN DE DATOS SIMULADOS

Una vez calibrados los valores de umbral para el sensor se procede a crear nuevos datos teniendo en cuenta las distribuciones supuestas para cada color. Se genera un data set de 500 datos para los valores de desviación, el código fuente se encuentra en el anexo B.

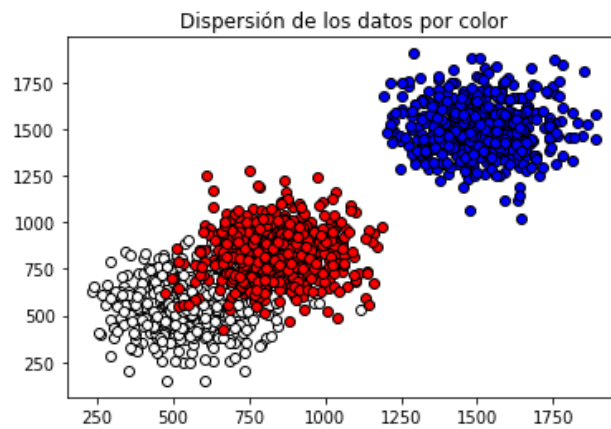


Ilustración 23. Dispersión de colores en el plano.²⁴

²⁴ Fuente: Autor

CAPITULO 3. MODELOS DE IA Y RESULTADOS

3.1.MODELOS DE MACHINE LEARNING

Para los datos tomados se realizará la aplicación de distintos modelos de clasificación, con el fin de verificar que tan buenos son estos modelos para clasificar dichos datos, el primero de los modelos que se utilizará será el de *Regresión Logística*.

3.1.1. Regresión Logística

La regresión logística es una técnica de análisis multivariante, en la que la variable dependiente o variable respuesta es una variable dicotómica y la variable o variables independientes pueden ser cualitativas o cuantitativas. Si en el modelo la variable independiente es cualitativa con H categorías, habrá que generar $H - 1$ variables DUMMY, A fin de que todas las posibilidades de la variable queden debidamente representadas en el modelo. (Alvares R.,1994)

La principal diferencia entre la regresión logística y la **regresión logística multinomial** es que la variable dependiente cualitativa no es dicotómica, sino que puede tener más de 2 categorías. (Guisante C.,2006)

Por sí solas, las regresiones logísticas son solo clasificadores binarios, lo que significa que no pueden manejar vectores objetivo con más de dos clases. Sin embargo, hay extensiones inteligentes para la regresión logística para hacer precisamente eso. En la **regresión logística de uno contra el resto (OVR)**, se entrena un modelo separado para cada clase que predice si una observación es esa clase o no (lo que lo convierte en un problema de clasificación binaria). Se supone que cada problema de clasificación (por ejemplo, clase 0 o no) es independiente. (Albon C.,2018)

Posterior a la aplicación y visualización de los resultados del modelo de regresión logística se aplican los modelos relacionados con las máquinas de vectores de soporte o SVM

3.1.2. SVM (Maquina de Vectores de Soporte)

Las Máquinas de Vector Soporte se fundamentan en el *Maximal Margin Classifier*, que, a su vez, se basa en el concepto de hiperplano.

hiperplano se define como un subespacio plano y afín de dimensiones $p-1$. El término afín significa que el subespacio no tiene por qué pasar por el origen. En un espacio de dos dimensiones, el hiperplano es un subespacio de 1 dimensión, es decir, una recta. En un espacio tridimensional, un hiperplano es un subespacio de dos dimensiones, un plano convencional. Para dimensiones $p > 3$ no es intuitivo visualizar un hiperplano, pero el concepto de subespacio con $p-1$ dimensiones se mantiene. (Amat J.,2017)

Cuando se dispone de n observaciones, cada una con p predictores y cuya variable respuesta tiene dos niveles (de aquí en adelante identificados como $+1$ y -1), se pueden emplear hiperplanos para construir un clasificador que permita predecir a que grupo pertenece una observación en función de sus predictores

- **Kernel lineal**

$$K(x, x') = x \cdot x'$$

Si se emplea un Kernel lineal, el clasificador Support Vector Machine obtenido es equivalente al Support Vector Classifier. (Amat J.,2017)

- **Kernel polinómico**

$$K(x, x') = (x \cdot x' + c) * d$$

Cuando se emplea $d=1$ y $c=0$, el resultado es el mismo que el de un kernel lineal. Si $d>1$, se generan límites de decisión no lineales, aumentando la no linealidad a medida que aumenta d . No suele ser recomendable emplear valores de d mayores 5 por problemas de overfitting. (Amat J.,2017)

- **Gaussian Kernel (RBF)**

$$K(x, x') = \exp(-\gamma ||x - x'||^2)$$

El valor de γ controla el comportamiento del kernel, cuando es muy pequeño, el modelo final es equivalente al obtenido con un kernel lineal, a medida que aumenta su valor, también lo hace la flexibilidad del modelo.

Debido a la utilización de *kernels*, en el ajuste de un SVM participa una matriz $n \times n$, donde n es el número de observaciones de entrenamiento. Por esta razón, lo que más influye en el tiempo de computación necesario para entrenar un SVM es el número de observaciones, no el de predictores. (Amat J.,2017)

Por último, se verificará la aplicación de modelos con un grado de complejidad superior como lo son:

3.1.3. Clasificador de K Vecinos Cercanos

El principio detrás de los métodos del vecino más cercano es encontrar un número predefinido de muestras de entrenamiento más cercanas en distancia al nuevo punto y predecir la etiqueta a partir de ellas. El número de muestras puede ser una constante definida por el usuario (k-aprendizaje vecino más cercano), o puede variar según la densidad local de puntos (aprendizaje vecino basado en el radio)

La clasificación basada en los vecinos es un tipo de aprendizaje basado en instancias o aprendizaje no generalizador: no intenta construir un modelo interno general, sino que simplemente almacena instancias de los datos de capacitación. La clasificación se calcula a partir de un voto de mayoría simple de los vecinos más cercanos de cada punto: a un punto de consulta se le asigna la clase de datos que tiene más representantes dentro de los vecinos más cercanos del punto. (Pedregosa F.,2011)

3.1.4. Clasificador de Procesos Gaussianos (Gaussian Process Classifier)

Los Modelos Naive Bayes con predictores gaussianos son modelos suponen que las observaciones tienen alguna distribución multivariada dada la pertenencia a la clase, pero el predictor o las entidades que componen la observación son independientes. Este marco de trabajo puede acomodar un conjunto de características completo de modo que una observación sea un conjunto de recuentos multinomiales.

3.1.5. Clasificador de Árboles de Decisión

Los árboles de decisión (DT) son un método de aprendizaje supervisado no paramétrico utilizado para la clasificación y la regresión. El objetivo es crear un modelo que prediga el valor de una variable objetivo mediante el aprendizaje de

reglas de decisión simples inferidas de las características de los datos. (Pedregosa F.,2011)

3.1.6. Clasificador de Bosque Aleatorio

Un bosque aleatorio es una colección de árboles de clasificación o clasificadores de árboles. Cada árbol se entrena en una muestra de arranque de los datos de entrenamiento, y en cada nodo de cada árbol, el algoritmo solo busca en un subconjunto aleatorio de las características para determinar una división. Para clasificar un vector de entrada en un bosque aleatorio, el vector se envía como entrada a cada uno de los árboles del bosque. Cada árbol da una clasificación, y se dice que el árbol vota por esa clase. En la clasificación, el bosque elige la clase que tiene más votos (sobre todos los árboles en el bosque). (C.H. Chen,2007)

3.1.7. Clasificador de Perceptrón Multicapa

Una red MLP está formada por un conjunto de procesadores elementales denominados neuronas, distribuidos en capas totalmente interconectadas. Cada elemento es una capa está conectado a las salidas de todos los elementos de la capa anterior. La función de transferencia de una neurona puede modelarse en dos etapas: la primera que da la activación como una suma ponderada de las entradas y una segunda que genera la salida como una función no lineal de la activación. Las redes MLP son muy flexibles para construir funciones de decisión ajustando los parámetros de conexión o pesos. (De la fuente M.J.,1999)

3.1.8. Clasificador Bayesiano Ingenuo

Los métodos ingenuos de Bayes son un conjunto de algoritmos de aprendizaje supervisado basados en la aplicación del teorema de Bayes con el supuesto "ingenuo"

de independencia condicional entre cada par de características dado el valor de la variable de clase. Los diferentes clasificadores ingenuos de Bayes difieren principalmente por los supuestos que hacen con respecto a la distribución de $P(x_i/y)$. (Pedregosa F.,2011)

3.2.CRITERIOS DE EVALUACIÓN PARA LOS MODELOS.

Para determinar si un modelo de Clasificación es bueno o no clasificando, se contará con las siguiente métricas o indicadores.

3.2.1. Precisión

La precisión del modelo está dada por la cantidad de valores positivos con relación al total de predicciones positivas que se realizaron. En otras palabras, la precisión es la capacidad que tiene un clasificador de no calificar una verdadera negativa como positiva.

Responde a la pregunta “Cuando se predicen valores positivos, ¿qué porcentaje clasifica correctamente?”, está dado por la siguiente ecuación:

$$Precision = \frac{Verdaderos\ Positivos}{Total\ Calificados\ Positivamente}$$

$$Precision = \frac{Verdaderos\ Positivos}{Verdaderos\ Positivos + Falsos\ Positivos}$$

3.2.2. Sensibilidad

Por otro lado, *la sensibilidad* es la capacidad que tiene el modelo clasificador de encontrar ejemplos positivos. Responde a la pregunta, “Cuando la clase es positiva, ¿qué porcentaje logra clasificar?”, está dado por la siguiente ecuación

$$Recall = \frac{Verdaderos\ Positivos}{Total\ Positivos}$$
$$Recall = \frac{Verdaderos\ Positivos}{Verdaderos\ Positivos + Falsos\ Negativos}$$

3.2.3. F1 Score

El valor F1 Se utiliza para combinar las medidas de *Precision* y *Recall* en un solo valor. El valor se calcula hallando la media armónica entre la precisión y la sensibilidad (Recall). Se expresa de la siguiente manera:

$$Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

El valor F1 facilita la interpretación del rendimiento debido a que considera los dos indicadores previamente mencionados por lo que se utilizara este parámetro para determinar que tan buenos son los modelos descritos en el apartado anterior.

3.3.APLICACIÓN DE MODELOS

Se han probado los trece algoritmos de machine learning en los datos de color dispersos el cual consta de una población total de 500 individuos repartidos uniformemente en las tres

categorías, se evalúa que tan buenos son los modelos para clasificar estos datos teniendo en cuenta el valor de *score* para cada uno de ellos.

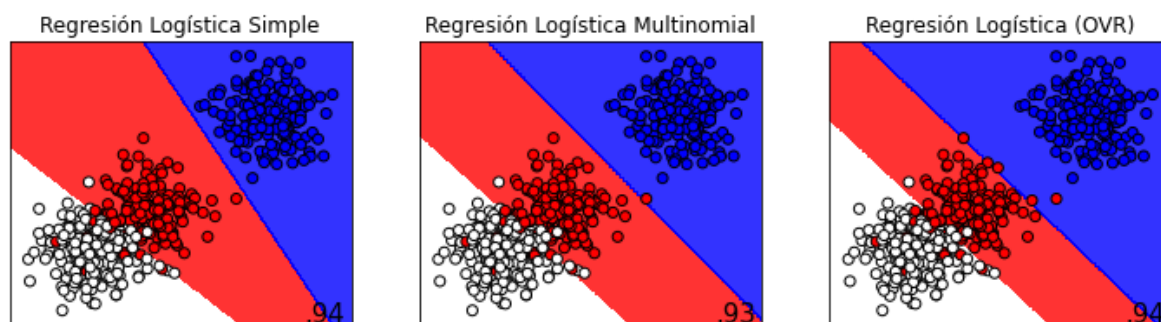


Ilustración 24. Resultados para regresión logística.²⁵

Al observar los resultados para la *regresión logística*, se obtienen resultados superiores al 90% de score, notando una leve mejoría en la predicción cuando se analiza con la regresión logística *simple* y en la regresión logística con OVR comparadas con la regresión logística multinomial.

²⁵ Fuente: Autor

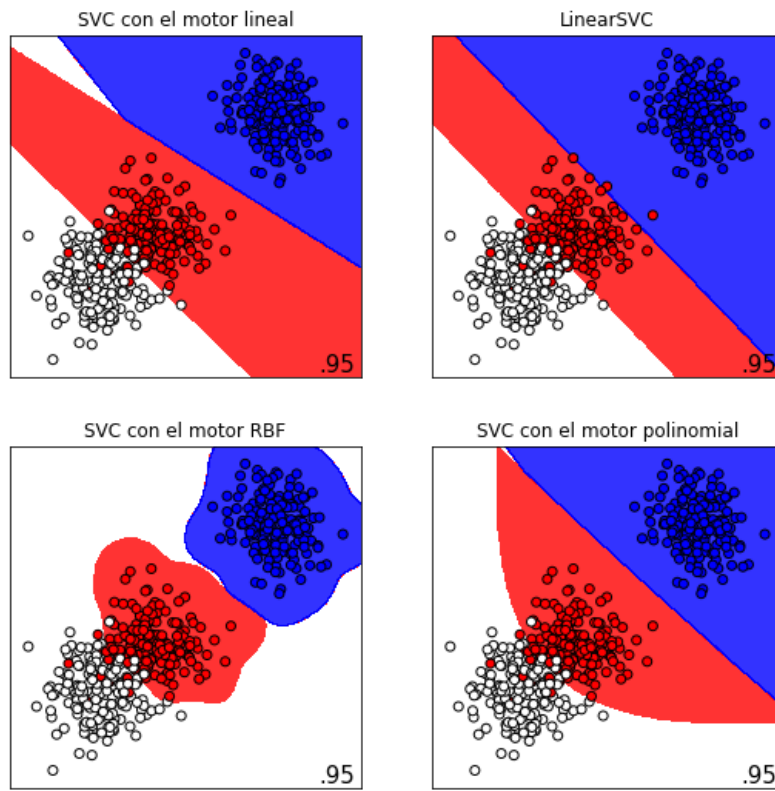


Ilustración 25. Resultados para SVC.²⁶

Al observar los resultados para los algoritmos de clasificación con vectores de soporte o SVC, se observa que dichos modelos arrojan mejores resultados que la regresión logística pero no se tiene una forma clara de discriminar cuál de ellos se ajusta mejor a los datos debido a que aparentemente todas arrojan el mismo valor de *score* el cual corresponde al .95, esto significa que ambos tipos de modelos son buenos para clasificar y predecir los datos, pero esto no necesariamente quiere decir que el modelo esté bien entrenado.

²⁶ Fuente: Autor

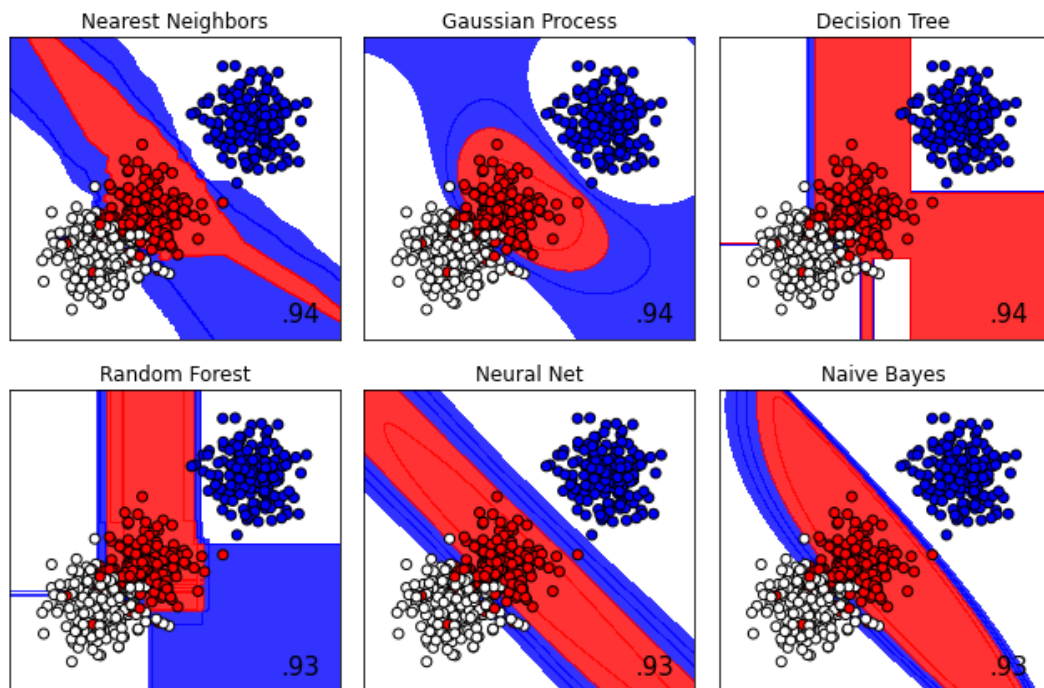


Ilustración 26. Resultados de modelos complejos.²⁷

Por último, se aplican modelos que tienen un mayor grado de complejidad y se observa que sus resultados presentan un nivel de score cercano a los modelos anteriormente planteados a pesar de que la forma en la que realizan su entrenamiento es diferente en cada uno de ellos. Se observa también, que los modelos que mejor se ajustan a los datos son los relacionados con las máquinas de vectores de soporte.

²⁷ Fuente: Autor

CONCLUSIONES

- El módulo clasificador de color de FischerTechnik cumple con las condiciones técnicas necesarias para la implementación de algoritmos de machine learning, esto debido a que se cuenta con un sensor de tipo analógico el cual permite extraer datos con cierto nivel de complejidad, que a la hora de aplicar los algoritmos de machine learning favorecen su implementación y su posterior análisis a través de los resultados obtenidos.
- La implementación del sensor de color Fischertechnik en la construcción del robot clasificador por color hizo posible la recopilación de datos de carácter tanto continuos como categóricos los cuales cumplen con los criterios necesarios para la implementación de los modelos de machine learning.
- Los modelos de machine learning aplicados a los datos presentan un buen nivel de ajuste posterior a su etapa de entrenamiento, no obstante, esto no significa que los modelos sean buenos prediciendo clasificaciones con nuevos datos, sino que son buenos prediciendo clases dentro de rangos ya conocidos para las distribuciones de color.
- Debido a la presente crisis sanitaria en relación con el COVID-19 no se pudo realizar mediciones del modelo dentro del laboratorio y mucho menos se pudo experimentar con la interacción de los modelos de machine learning en la programación del robot clasificador por color por lo que se imposibilita determinar si existe o no diferencia entre las predicciones realizadas por los modelos que han sido alimentados con los datos simulados y los que han sido alimentados con los datos tomados del módulo de FischerTechnik.

RECOMENDACIONES

- Se recomienda tener en cuenta aspectos como la hora del día o la cantidad de radiación lumínica natural, ya que esta puede intervenir en las mediciones del sensor, es necesario asegurar que los datos tomados desde el módulo de FischerTechnik se hagan bajo un ambiente controlado, o que las variables que intervienen y/o interfieren en la medición se encuentren controladas.
- Observando la capacidad de integración que tienen las herramientas del lenguaje Python mediante la librería ftrobopy, es posible realizar rutinas inteligentes como el calibrador que se ejecuten en tiempo real mientras el sistema está operando y realicen su labor (calibrar/mejorar el proceso) a lo largo de la operación de este mismo.
- Por lo que es posible que al implementar los modelos de machine learning tratados en el presente trabajo investigativo con la programación a través de ftrobopy, estos modelos puedan interactuar con el funcionamiento del módulo en tiempo real.

BIBLIOGRAFIA

ADEC, A. para el desarrollo tecnológico de la ciudad de córdoba. (2017). Los empleos del futuro, 1–6.

BARRIENTOS, Antonio. Fundamentos de Robótica 1a Edición. McGraw Hill, 1997.

BERNAL, Cesar. Metodología de la investigación. cuarta edición, Pearson, Colombia, 2016.

CHASE, R.B. and AQUILANO, N.J. Administración de operaciones, Producción y cadena de suministros, 13a Edición. McGraw Hill, 2014.

Enríquez, G. (2004). El ABC Del Alumbrado Y Las Instalaciones Eléctricas En Baja Tensión / the ABC's of Lighting and Low Tensión Eléctrica Instalaciones.

Ernst & Young LLP. (2019). Emerging Technologies: Changing how we live, work and play, 1–68. Recuperado de [https://www.ey.com/Publication/vwLUAssets/ey-mint-emerging-technologies-report-2019/\\$File/ey-mint-emerging-technologies-report-2019.pdf](https://www.ey.com/Publication/vwLUAssets/ey-mint-emerging-technologies-report-2019/$File/ey-mint-emerging-technologies-report-2019.pdf)

Fundación de la innovación Bankinter. (2011). El internet de las cosas. Internet de las cosas: En un mundo conectado de objetos inteligentes, (pág. 78).

GROOVER, P, Mikel, Automation Production Systems, 3a Edición. McGraw Hill, 2008.

Lucas, R. E. (1996). LA REVOLUCIÓN INDUSTRIAL: PASADO Y FUTURO. Estudios Públicos (Vol. 64).

Matich, D. J. (2001). *Redes Neuronales: Conceptos Básicos y Aplicaciones* . Rosario: Universidad Tecnológica Nacional - Facultad Regional Rosario.

Martín, J, García, M. (2016). Motores eléctricos (Automatismos industriales).

Moreno, A., Armengol, E., Béjar, J., Belanche, L., Cortés, U., Gavalda, R., . . . Sánchez, M. (1998). *Aprendizaje Automático*. Cataluña: Edicions UPC.

NAHMIAS, S. *Análisis de la Producción y las Operaciones*. 6a Edición. McGraw Hill, 2014.

OLANREWAJU, Tunde, WILLMOTT, Paul. Finding your digital sweet spot. Artículo recuperado de <https://www.mckinsey.com>, noviembre 2013.

Overton, R. C. (1952). The Transportation Revolution, 1815–1860. By George Rogers Taylor. [Economic History of the United States, Volume IV.] (New York: Rinehart and Company. 1951. Pp. xvii, 490. \$4.50.). *The American Historical Review*, 57(3), 701–703. <https://doi.org/10.1086/ahr/57.3.701>

Ostos, P. (2012). UF1027 - Caracterización y selección del equipamiento frigorífico.

Rao, S. K., & Prasad, R. (2018). Impact of 5G Technologies on Industry 4.0. *Wireless Personal Communications*, 100(1), 145–159. <https://doi.org/10.1007/s11277-018-5615-7>

Rodríguez Jiménez, Andrés; Pérez Jacinto, Alipio Omar (2017). Métodos científicos de indagación y de construcción de conocimiento. *Revista de escuela de Administración de Negocios. Universidad EAN*. Bogotá, Colombia.

Roldán, J. (2012). Tecnología y circuitos de aplicación de neumática hidráulica y electricidad.

Rosenblatt, F. (1958). El perceptrón: Un modelo probabilístico para el almacenamiento de la información en el cerebro. *Revisión Psicológica*, 386 - 408.

Vallejo, G. de O. (1992). *Técnicas multivariadas aplicadas a las ciencias del comportamiento* - Guillermo Vallejo Seco - Google Libros.